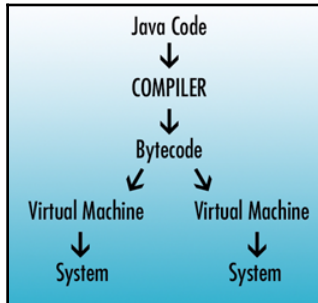


Chapter 1: Getting Started with Java



Java SE Downloads



Java Platform (JDK) 8u144



NetBeans with JDK 8














Java Platform, Standard Edition

Java SE 8u144
Java SE 8u144 includes important bug fixes. Oracle strongly recommends that all Java SE 8 users upgrade to this release.
[Learn more](#) ▶

<ul style="list-style-type: none">Installation InstructionsRelease NotesOracle LicenseJava SE ProductsThird Party LicensesCertified System ConfigurationsReadme Files<ul style="list-style-type: none">JDK ReadMeJRE ReadMe	<p>JDK</p> <p>DOWNLOAD ↓</p> <p>Server JRE</p> <p>DOWNLOAD ↓</p> <p>JRE</p> <p>DOWNLOAD ↓</p>
--	--

Java SE Development Kit 8u144 Demos and Samples Downloads

You must accept the [Oracle BSD License](#). to download this software.
Thank you for accepting the Oracle BSD License.; you may now download this software.

Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	9.95 MB	 jdk-8u144-linux-arm32-vfp-hflt-demos.tar.gz
Linux ARM 64 Hard Float ABI	9.94 MB	 jdk-8u144-linux-arm64-vfp-hflt-demos.tar.gz
Linux x86	52.66 MB	 jdk-8u144-linux-i586-demos.rpm
Linux x86	52.52 MB	 jdk-8u144-linux-i586-demos.tar.gz
Linux x64	52.72 MB	 jdk-8u144-linux-x64-demos.rpm
Linux x64	52.54 MB	 jdk-8u144-linux-x64-demos.tar.gz
Mac OS X	53.09 MB	 jdk-8u144-macosx-x86_64-demos.zip
Solaris x64	13.52 MB	 jdk-8u144-solaris-x64-demos.tar.Z
Solaris x64	9.31 MB	 jdk-8u144-solaris-x64-demos.tar.gz
Solaris SPARC 64-bit	13.58 MB	 jdk-8u144-solaris-sparcv9-demos.tar.Z
Solaris SPARC 64-bit	9.34 MB	 jdk-8u144-solaris-sparcv9-demos.tar.gz
Windows x86	53.8 MB	 jdk-8u144-windows-i586-demos.zip
Windows x64	53.82 MB	 jdk-8u144-windows-x64-demos.zip

NetBeans | NetBeans IDE | NetBeans Platform | Plugins | Docs & Support | Community | Partners | Search

HOME / Download

NetBeans IDE 8.2 Download

8.1 | 8.2 | Development | Archive

Email address (optional):

Subscribe to newsletters: Monthly Weekly

NetBeans can contact me at this address

IDE Language: **English** Platform: **Windows**

Note: Greyed out technologies are not supported for this platform.

NetBeans IDE Download Bundles

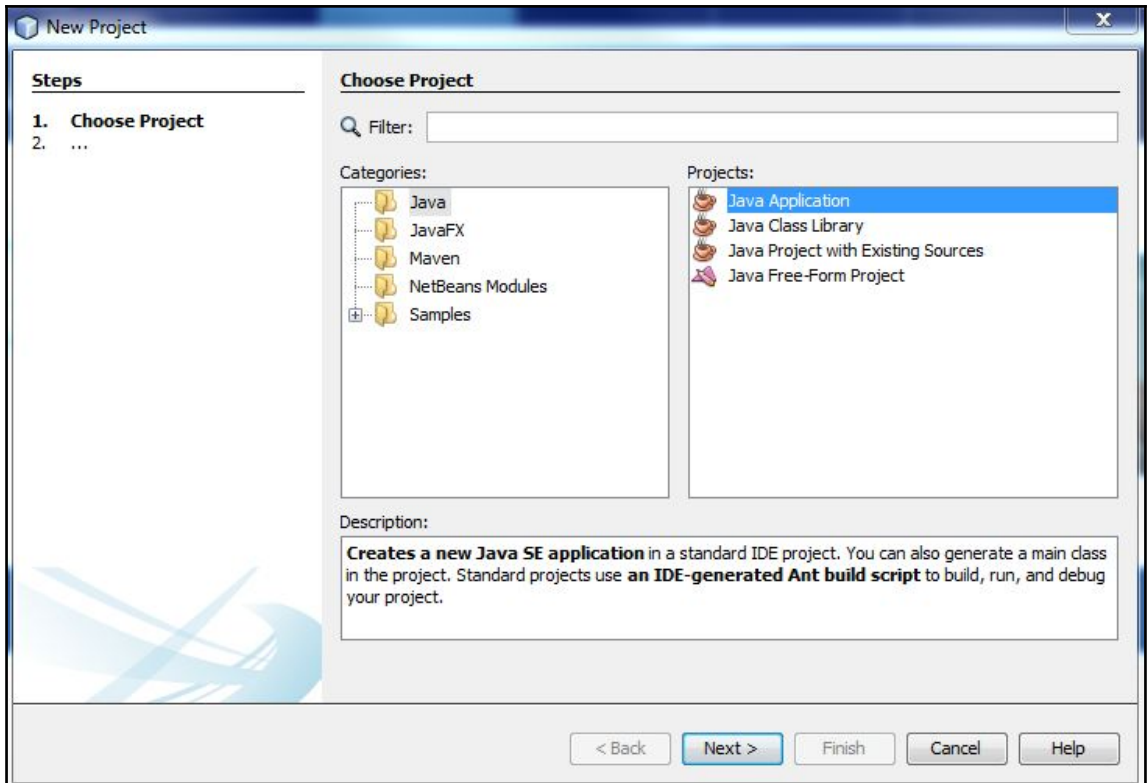
Supported technologies *	Java SE	Java EE	HTML5/JavaScript	PHP	C/C++	All
NetBeans Platform SDK	•	•				•
Java SE	•	•				•
Java FX	•	•				•
Java EE		•				•
Java ME						•
HTML5/JavaScript		•	•	•		•
PHP			•	•		•
C/C++					•	•
Groovy						•
Java Card™ 3 Connected						•
Bundled servers						
GlassFish Server Open Source Edition 4.1.1		•				•
Apache Tomcat 8.0.27		•				•

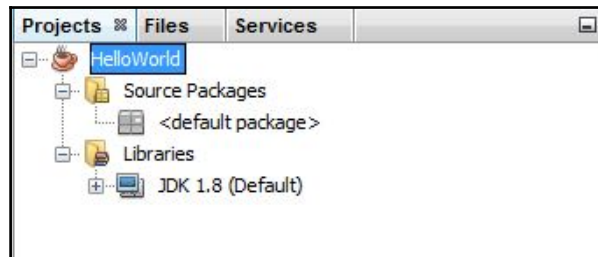
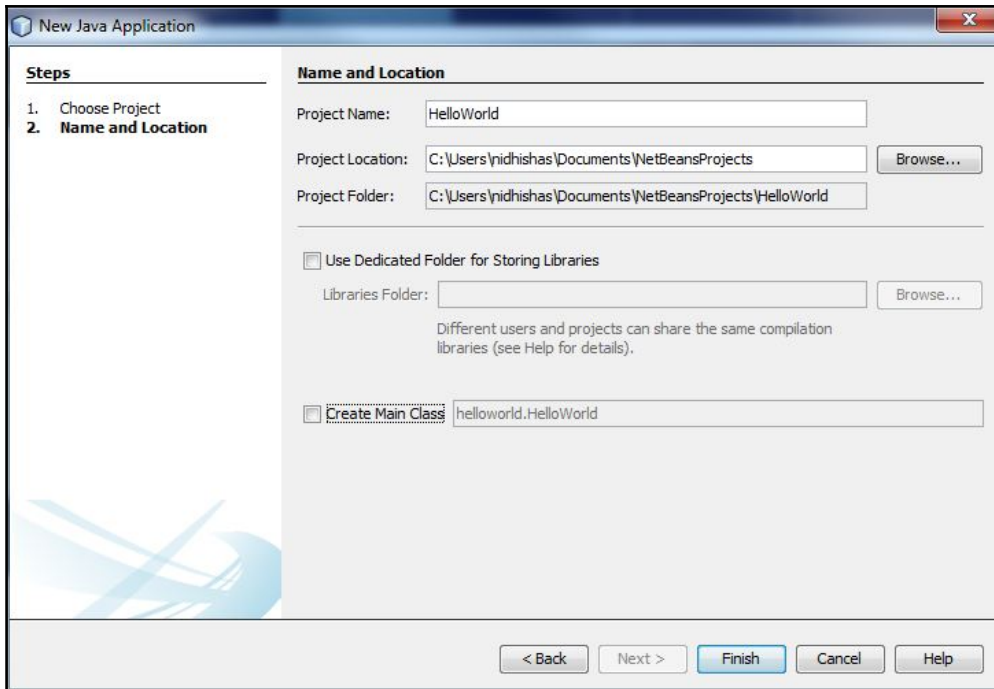
Download buttons: Download, Download, Download x86, Download x86, Download x86, Download x64, Download x64, Download x64, Download

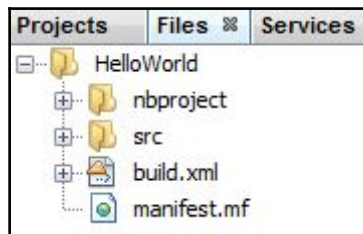
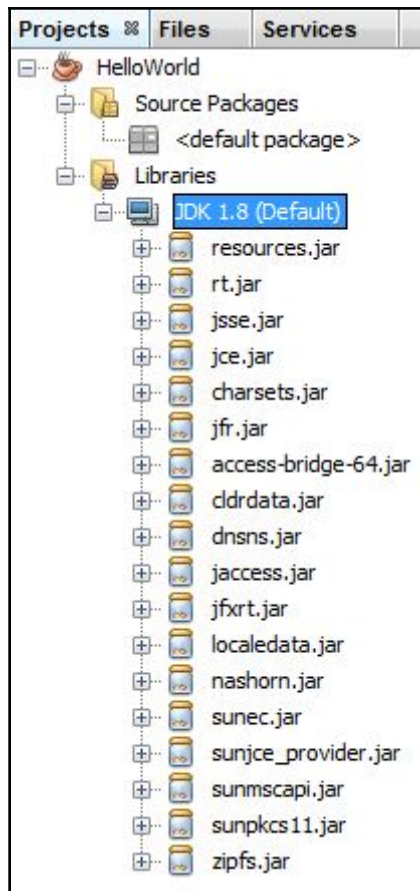
Free, 95 MB Free, 197 MB Free, 108 - 112 MB Free, 108 - 112 MB Free, 107 - 110 MB Free, 221 MB

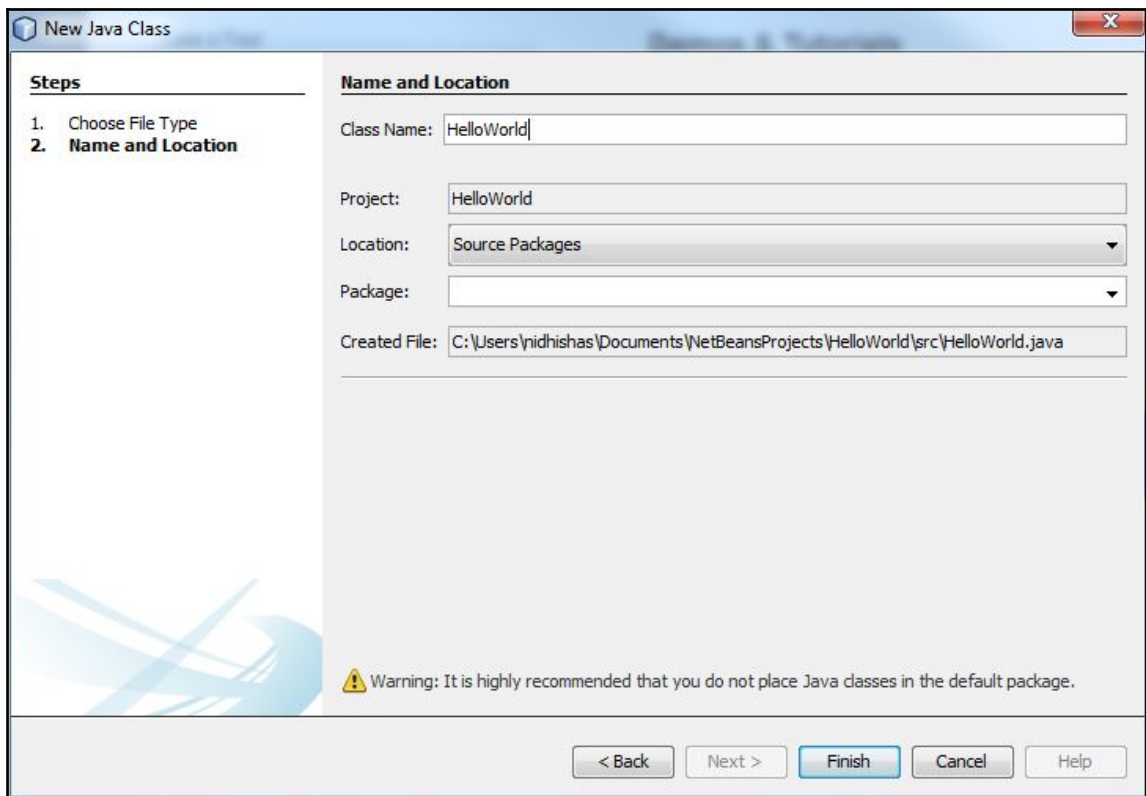
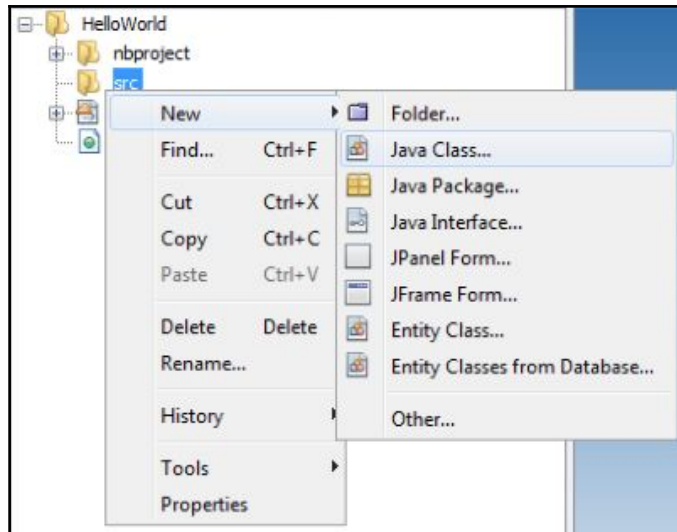
NetBeans IDE 8.2 Download Started

 Your download should start automatically.
If not, [download it here.](#)











```
1  /*
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6
7  /**
8  *
9  * @author nidhishas
10 */
11 public class HelloWorld {
12
13 }
14
```

```
1
2 public class HelloWorld {
3     public static void main(String[] args) {
4         println("Hello World!");
5     }
6 }
7
```

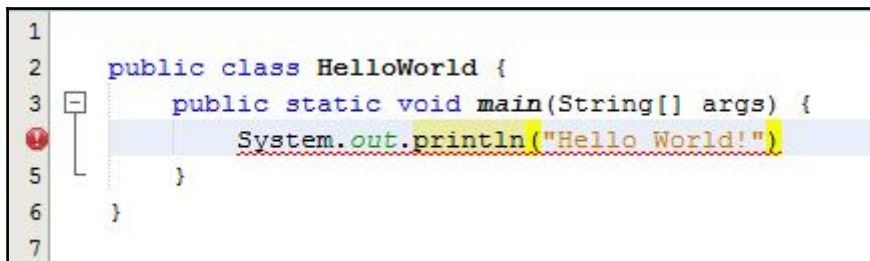
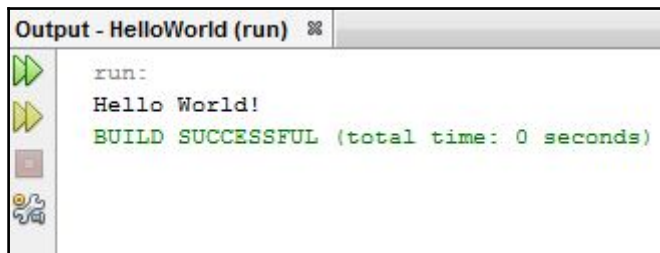
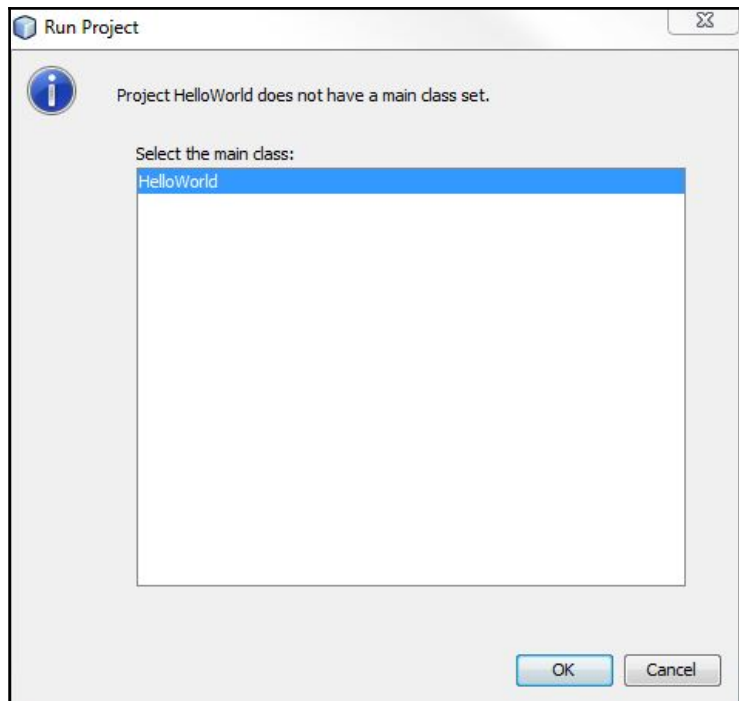
```
1
2 public class HelloWorld {
3     public static void main(String[] args) {
4         println("Hello World!");
5     }
6 }
7
```



```
1  
2 public class HelloWorld {  
3     public static void main(String[] args) {  
4         System.out.println("Hello World!");  
5     }  
6 }  
7
```

Output - HelloWorld (clean.jar)  
deps-jar:
Created dir: C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\build
Updating property file: C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\build\build-jar.properties
Created dir: C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\build\classes
Created dir: C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\build\empty
Created dir: C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\build\generated-sources\ap-source-output
Compiling 1 source file to C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\build\classes
compile:
Created dir: C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\dist
Copying 1 file to C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\build
Nothing to copy.
Building jar: C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\dist\HelloWorld.jar
To run this application from the command line without Ant, try:
java -jar "C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\dist\HelloWorld.jar"
jar:
BUILD SUCCESSFUL (total time: 0 seconds)

```
1  
2 public class HelloWorld {  
3     public static void main(String[] args) {  
4         System.out.println("Hello World!");  
5     }  
6 }  
7
```



```
Output - HelloWorld (cleanjar)
Deleting directory C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\build
clean:
init:
deps-jar:
Created dir: C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\build
Updating property file: C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\build\build-jar.properties
Created dir: C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\build\classes
Created dir: C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\build\empty
Created dir: C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\build\generated-sources\ap-source-output
Compiling 1 source file to C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\build\classes
C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\src\HelloWorld.java:4: error: ';' expected
    System.out.println("Hello World!")
    ^
1 error
C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\nbproject\build-impl.xml:930: The following error occurred while executing this line:
C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\nbproject\build-impl.xml:270: Compile failed; see the compiler error output for details.
BUILD FAILED (total time: 0 seconds)
```

```
1
2 public class HelloWorld {
3     public static void main(String[] args) {
4         System.out.println("Hello World");
5     }
6 }
7
```

```
Output - HelloWorld (jar)
Updating property file: C:\Users\Zachary\Documents\NetBeansProjects\LTP\HelloWorld\build\build-jar.properties
Compiling 1 source file to C:\Users\Zachary\Documents\NetBeansProjects\LTP\HelloWorld\build\classes
C:\Users\Zachary\Documents\NetBeansProjects\LTP\HelloWorld\src\HelloWorld.java:4: error: ';' expected
    System.out.println("Hello World")
    ^
1 error
C:\Users\Zachary\Documents\NetBeansProjects\LTP\HelloWorld\nbproject\build-impl.xml:926: The following error occurred
C:\Users\Zachary\Documents\NetBeansProjects\LTP\HelloWorld\nbproject\build-impl.xml:268: Compile failed; see the compi
```

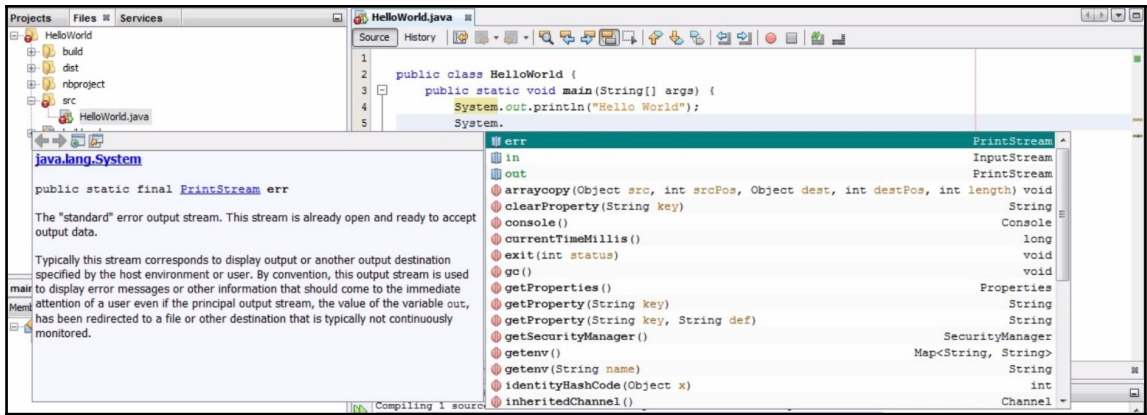
```
1
2 public class HelloWorld {
3     public static void main(String[] args) {
4         System.out.println("Hello World!");
5     }
6 }
7
```

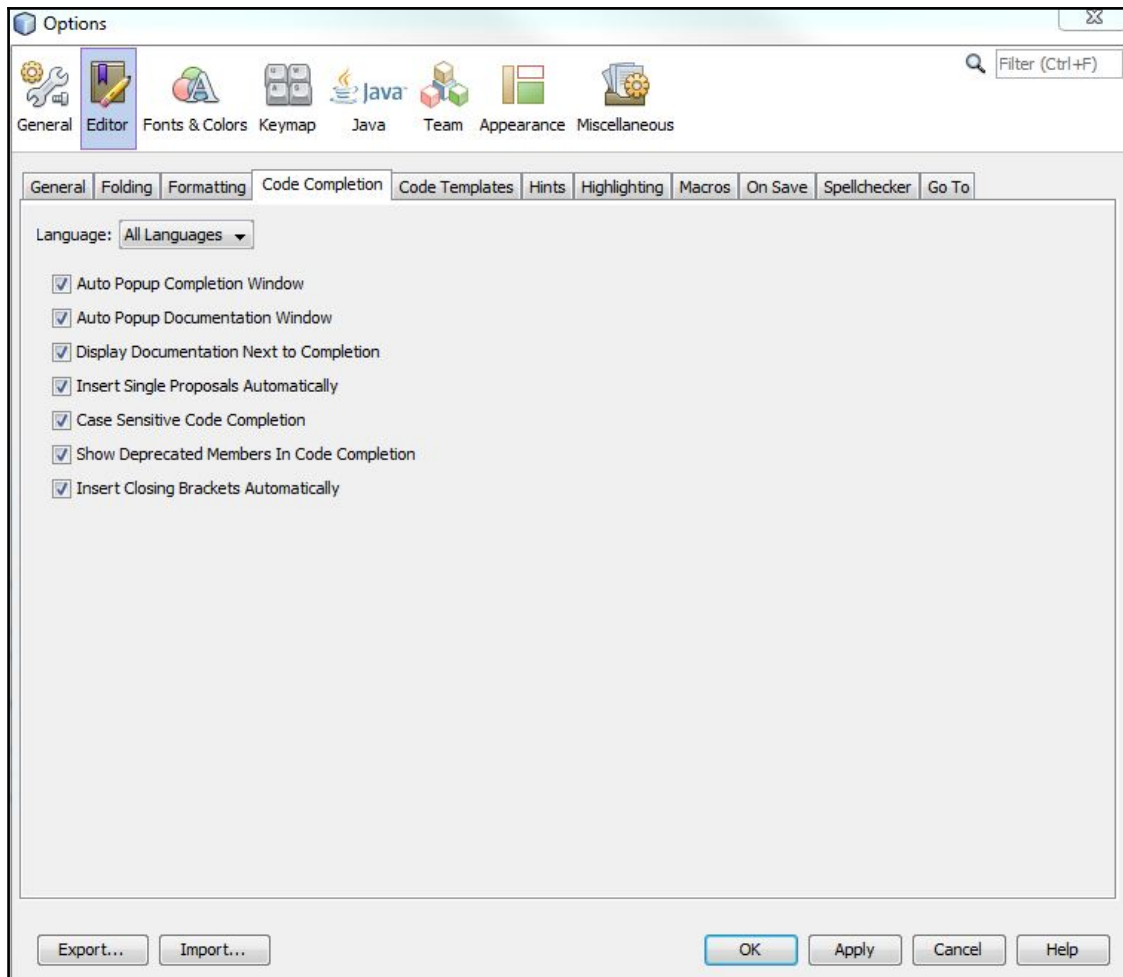
Output - HelloWorld (jar)

```
ant -f C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld -Dnb.internal.action.name=build jar
init:
Deleting: C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\build\build-jar.properties
deps-jar:
Updating property file: C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\build\build-jar.properties
Compiling 1 source file to C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\build\classes
compile:
Created dir: C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\dist
Copying 1 file to C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\build
Nothing to copy.
Building jar: C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\dist\HelloWorld.jar
To run this application from the command line without Ant, try:
java -jar "C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\dist\HelloWorld.jar"
jar:
BUILD SUCCESSFUL (total time: 0 seconds)
```

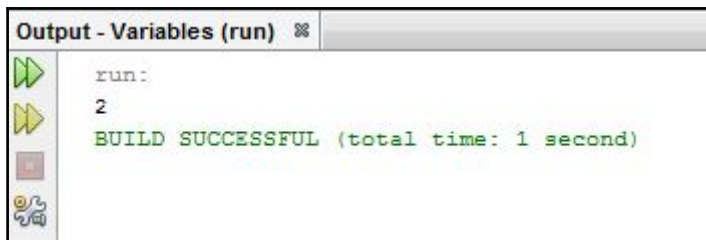
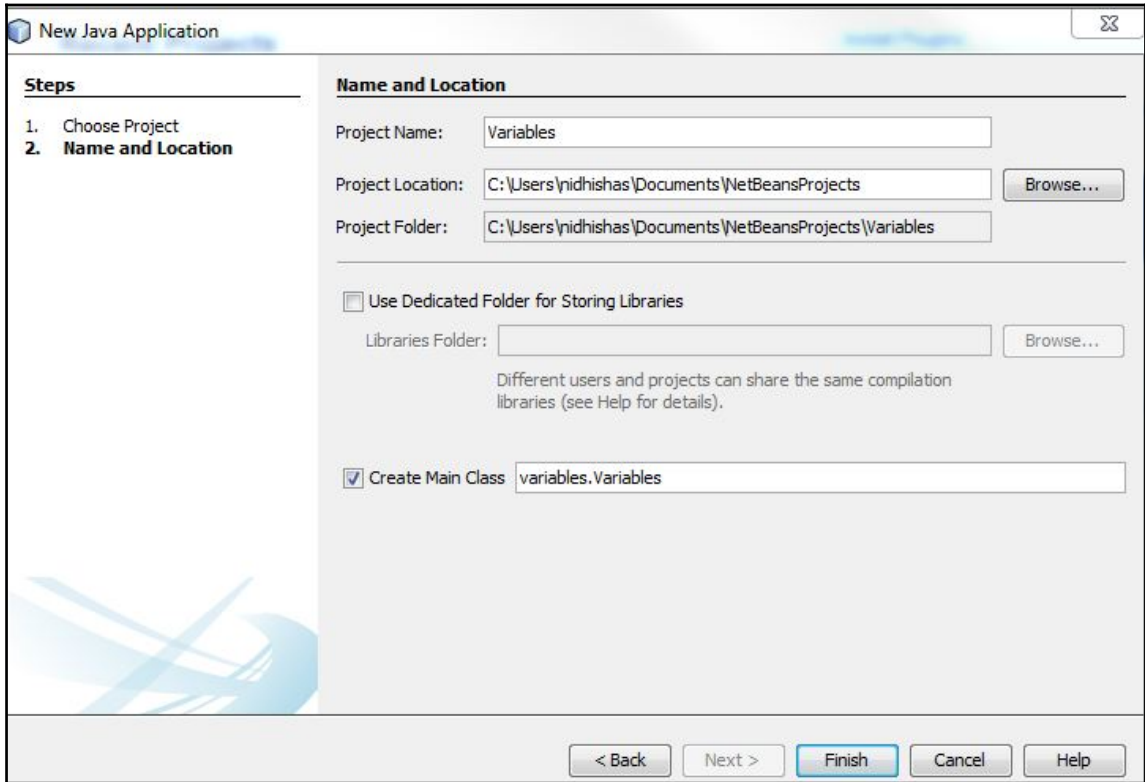
```
1
2 public class HelloWorld {
3     public static void main(String[] args) {
4         System.out.println("Hello World!");
5     }
6 }
7
```

```
Compiling 1 source file to C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\build\classes
C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\src\HelloWorld.java:4: error: not a statement
    System.out.println("Hello World!");
    ^
C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\src\HelloWorld.java:4: error: ';' expected
    System.out.println("Hello World!");
    ^
2 errors
C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\nbproject\build-impl.xml:930: The following error occurred while executing this line:
C:\Users\nidhishas\Documents\NetBeansProjects\HelloWorld\nbproject\build-impl.xml:270: Compile failed; see the compiler error output for details.
BUILD FAILED (total time: 0 seconds)
```





Chapter 2: Understanding Typed Variables



```
Output - Variables (run) %
run:
5
-1
6
0
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
Output - Variables (run) %
run:
5
-1
6
0
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
3 public class Variables {
4
5 public main(String[] args) {
6     int x = 5;
7     int x = -1;
8     int x = 6;
9     int x = 0;
10 }
```

';' expected
variable x is already defined in method main(String[])

(Alt-Enter shows hints)

int x

```
Output - Variables (run) %
run:
7
1
12
1
BUILD SUCCESSFUL (total time: 0 seconds)
```



```
Output - Variables (run) %
run:
-5
-15
-50
-2
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
3 public class Variables {
4
5 public static void main(String[] args) {
6     int
7     x = 2147483648;
9
10 //2147483647
11 //-2147483648
12
13 System.out.println(x);
14 }
15 }
```

```
Output - Variables (run) %
run:
-2147483648
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
Output - Variables (run) %
run:
2147483648
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
float fNumber = 5.5;
```

```
Output - FloatingPointNumbers (jar)
ant -f C:\Users\nidhishas\Documents\NetBeansProjects\FloatingPointNumbers -Dnb.internal.action.name=build_jar
init:
depur:jar:
Created dir: C:\Users\nidhishas\Documents\NetBeansProjects\FloatingPointNumbers\build
Updating property file: C:\Users\nidhishas\Documents\NetBeansProjects\FloatingPointNumbers\build\build-jar.properties
Created dir: C:\Users\nidhishas\Documents\NetBeansProjects\FloatingPointNumbers\build\classes
Created dir: C:\Users\nidhishas\Documents\NetBeansProjects\FloatingPointNumbers\build\empty
Created dir: C:\Users\nidhishas\Documents\NetBeansProjects\FloatingPointNumbers\build\generated-sources\ap-source-output
Compiling 1 source file to C:\Users\nidhishas\Documents\NetBeansProjects\FloatingPointNumbers\build\classes
C:\Users\nidhishas\Documents\NetBeansProjects\FloatingPointNumbers\src\FloatingPointNumbers\FloatingPointNumbers.java:10: error: incompatible types: possible lossy conversion from double to float
    float fNumber = 5.5;
    ^
1 error
C:\Users\nidhishas\Documents\NetBeansProjects\FloatingPointNumbers\vbproject\build-impl.xml:980: The following error occurred while executing this line:
C:\Users\nidhishas\Documents\NetBeansProjects\FloatingPointNumbers\vbproject\build-impl.xml:270: Compile failed; see the compiler error output for details.
BUILD FAILED (total time: 1 second)
```

```
Output - FloatingPointNumbers (run)
run:
5
5.5
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
Output - FloatingPointNumbers (run)
run:
1
1.25
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
Output - FloatingPointNumbers (run)
run:
1
1.0
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
Output - FloatingPointNumbers (run)
run:
1
1.25
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
Output - FloatingPointNumbers (run) %
run:
0.0
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
Output %
Debugger Console % FloatingPointNumbers (run) %
run:
0.8333333
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
7 double dNumber1 = 5.7;
8 double dNumber2 = 6.85431;
float fNumber = dNumber1/dNumber2;
```

```
Output %
Debugger Console % FloatingPointNumbers (run) %
run:
0.8315936
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
Output %
Debugger Console % TheMathLib (run) %
run:
4.321
BUILD SUCCESSFUL (total time: 0 seconds)
```

Methods	
Modifier and Type	Method and Description
static double	abs(double a) Returns the absolute value of a double value.
static float	abs(float a) Returns the absolute value of a float value.
static int	abs(int a) Returns the absolute value of an int value.
static long	abs(long a) Returns the absolute value of a long value.
static double	acos(double a) Returns the arc cosine of a value; the returned angle is in the range 0.0 through π .
static double	asin(double a) Returns the arc sine of a value; the returned angle is in the range $-\pi/2$ through $\pi/2$.
static double	atan(double a) Returns the arc tangent of a value; the returned angle is in the range $-\pi/2$ through $\pi/2$.
static double	atan2(double y, double x) Returns the angle θ from the conversion of rectangular coordinates (x, y) to polar coordinates (r, θ).

static double	pow(double a, double b) Returns the value of the first argument raised to the power of the second argument.
---------------	--

```
float number = 4.321f;
number = pow;
```

```
5 double number = 4.321;
number = pow(number, 4.0);
```

Output

Debugger Console TheMathLib (run)

```
run:
348.6077720236809
BUILD SUCCESSFUL (total time: 0 seconds)
```

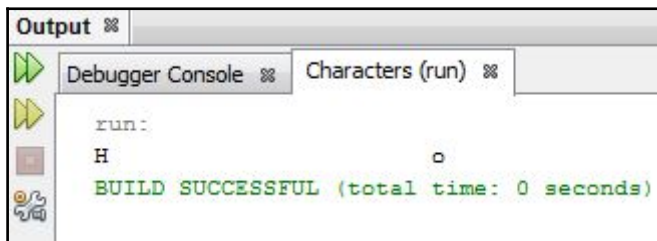
```
char character1 = 'Hi';
```

Output

Debugger Console Characters (run)

```
run:
Hello
BUILD SUCCESSFUL (total time: 0 seconds)
```

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040	 	Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	\$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	((72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051))	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[[123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174	|	
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135]]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177		DEL



```

package stringsinjava;

public class StringsInJava {
    public static void main(String[] args) {
        char c = 'c';
        String s = "string";
    }
}

```

```
Output %
Debugger Console % StringsInJava (run) %
run:
stringonestringtwoLIT
BUILD SUCCESSFUL (total time: 0 seconds)
```

String toUpperCase()
Converts all of the characters in this String to upper case using the rules of the default locale.

```
Output %
Debugger Console % StringsInJava (run) %
run:
STRINGONESTRINGTWOLIT
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
Output %
Debugger Console % StringsInJava (run) %
run:
STRINGONESTRINGTWOLIT
stringonestringtwoLIT
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
Output %
Debugger Console % StringsInJava (run) %
run:
STRINGONESTRINGTWOLIT
STRINGONESTRINGTWOLIT
BUILD SUCCESSFUL (total time: 0 seconds)
```

String replace(char oldChar, char newChar)
Returns a new string resulting from replacing all occurrences of oldChar in this string with newChar.

```

Output »
Debugger Console » StringsInJava (run) »
run:
STRINoONESTRINoTWOLIT
STRINGONESTRINGTWOLIT
BUILD SUCCESSFUL (total time: 0 seconds)

```

```

4 public static void main(String[] args) {
5     /*char c = 'c';
6     String s1 = "stringone";
7     String s2 = "stringtwo";
8     String s3 = s1 + s2 + "LIT";
9
10    s3 = s3.toUpperCase();
11
12    System.out.println( s3.replace
13    System.out.println( s3 );*/
14
15    String s = "The program says: |"Hello World"|";
16    System.out.println(s);
17 }

```

;' expected
 ;' expected

 (Alt-Enter shows hints)

```

Output »
Debugger Console » StringsInJava (run) »
run:
The program says: "Hello World"
BUILD SUCCESSFUL (total time: 0 seconds)

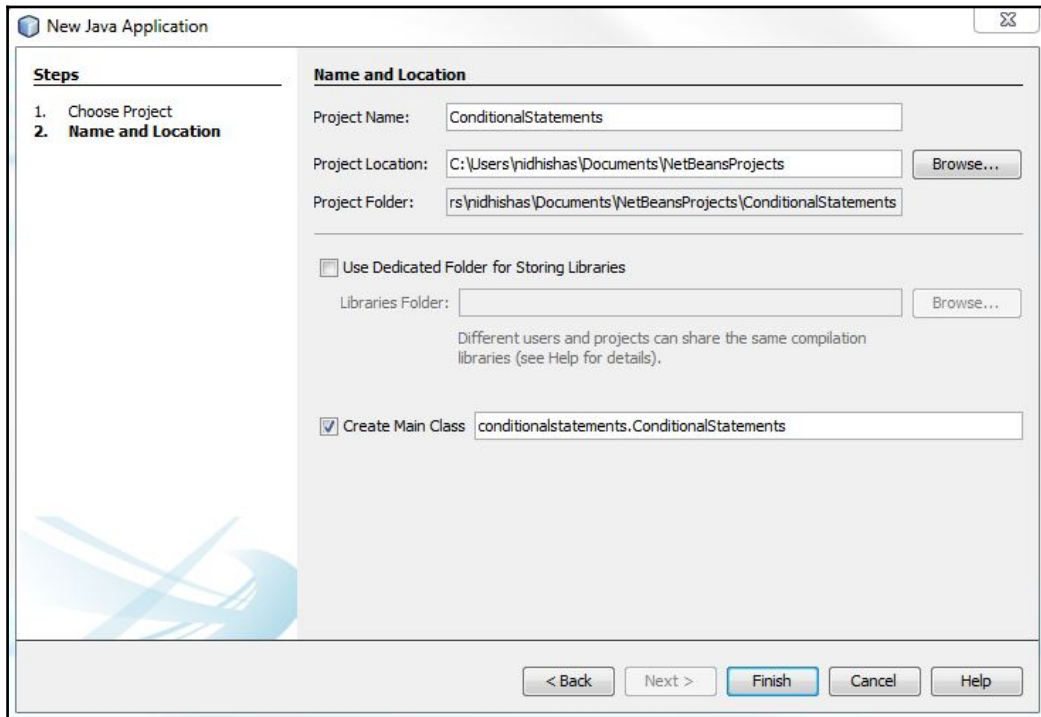
```


```


Output »
Debugger Console » StringsInJava (run) »
run:
The program says: \ "Hello World"
BUILD SUCCESSFUL (total time: 0 seconds)


```


Chapter 3: Branching




Output - ConditionalStatements (jar) %	Variables	Breakpoints
 <pre>ant -f C:\Users\nidhishas\Documents\NetBeansProjects\ConditionalStatements -Dnb.internal.action.name=build jar init: Deleting: C:\Users\nidhishas\Documents\NetBeansProjects\ConditionalStatements\build\build-jar.properties deps-jar: Updating property file: C:\Users\nidhishas\Documents\NetBeansProjects\ConditionalStatements\build\build-jar.properties compile: Copying 1 file to C:\Users\nidhishas\Documents\NetBeansProjects\ConditionalStatements\build Nothing to copy. To run this application from the command line without Ant, try: java -jar "C:\Users\nidhishas\Documents\NetBeansProjects\ConditionalStatements\dist\ConditionalStatements.jar" jar: BUILD SUCCESSFUL (total time: 0 seconds)</pre>		

Output - ConditionalStatements (run) %
 <pre>run: hi hi BUILD SUCCESSFUL (total time: 34 seconds)</pre>

Output - ConditionalStatements (run) #4 %	Variables	Breakpoints
 <pre>run: Input now: 15 MORE! BUILD SUCCESSFUL (total time: 1 second)</pre>		

Output - ConditionalStatements (run) #4 %	Variables	Breakpoints
 <pre>run: Input now: 10 LESS BUILD SUCCESSFUL (total time: 1 second)</pre>		

Output - ConditionalStatements (run) #4 %	Variables	Breakpoints
 <pre>run: Input now: 10 LESS BUILD SUCCESSFUL (total time: 2 seconds)</pre>		

```

10     System.out.println("Input now: ");
11     String input = reader.next();
12
    if(input == "password")

```

Output - ConditionalStatements (run) #5	Variables	Breakpoints
run:		
Input now:		
password		
YES		
BUILD SUCCESSFUL (total time: 4 seconds)		

Output - ComplexConditionals (run) #4	Variables	Breakpoints
run:		
abcdefg		
TRUE		
BUILD SUCCESSFUL (total time: 6 seconds)		

Output - ComplexConditionals (run) #2	Variables	Breakpoints
run:		
hgtreh6jt6		
FALSE		
BUILD SUCCESSFUL (total time: 12 seconds)		

Output - ComplexConditionals (run) #4	Variables	Breakpoints
run:		
abcz		
TRUE		
BUILD SUCCESSFUL (total time: 5 seconds)		





complexconditionals.ComplexConditionals > main > if (input.contains(sOne) input.contains(sTwo)) >		
Output - ComplexConditionals (run) #2	Variables	Breakpoints
run:		
zhhj95njho95nh		
TRUE		
BUILD SUCCESSFUL (total time: 59 seconds)		





Output - ComplexConditionals (run)	
run:	
zabc	
TRUE	
BUILD SUCCESSFUL (total time: 6 seconds)	





Output - ComplexConditionals (run)	
run:	
zabc	
FALSE	
BUILD SUCCESSFUL (total time: 9 seconds)	





Output - Switcher (run)	Variables	Breakpoints
run:		
RED		
BUILD SUCCESSFUL (total time: 0 seconds)		

Output	Variables	Breakpoints
introloops (run) #9		
Switcher (run)		
run:		
RED		
BLUE		
GREEN		
NONE		
BUILD SUCCESSFUL (total time: 0 seconds)		

Output - Switcher (run) %	Variables	Breakpoints
   	<pre>run: RED BUILD SUCCESSFUL (total time: 0 seconds) </pre>	

Output %	Variables	Breakpoints
   	introloops (run) #9 % Switcher (run) %	
	<pre>run: RED BUILD SUCCESSFUL (total time: 0 seconds) </pre>	

Output %	Variables	Breakpoints
   	introloops (run) #9 % introloops (run) #10 %	
	<pre>run: Hello world Hello world Hello world Hello world Hello world BUILD SUCCESSFUL (total time: 0 seconds)</pre>	

Output - introloops (run) #9 %	
   	<pre>Hello world Hello world Hello world Hello world Hello world Hello world</pre>

Output - introloops (run) #9	Variables	Breakpoints
run: welcome to loops in Java welcome to loops in Java BUILD SUCCESSFUL (total time: 12 seconds)		

```
while(!input.equals("STOP"))
```

Output - introloops (run) #9	Variables	Breakpoints
run: a b c d e f g h i j k l m n o p q r STOP a b c d e f g h i j k l m n o p q rSTOP BUILD SUCCESSFUL (total time: 38 seconds)		

```

package forloops;

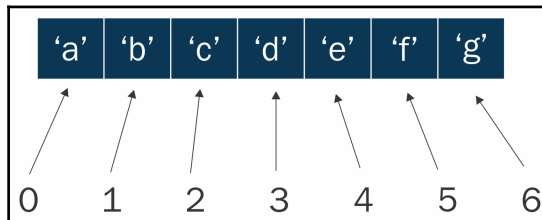
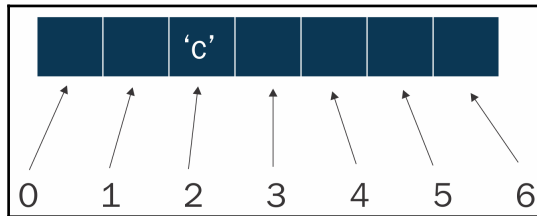
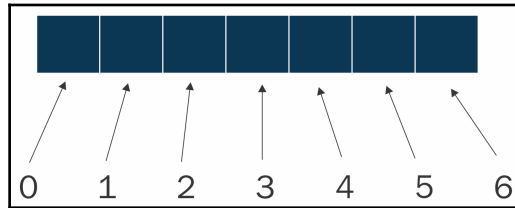
public class Forloops {
    public static void main(String[] args) {
        /* int i=1;
        while(i <= 100)
        {
            System.out.println(i);
            i++;
        }*/
    }
}

```




The screenshot shows an IDE window titled 'forloops.Forloops'. The main area displays the output of a build process. On the left side, there are several icons: a green double arrow, a yellow double arrow, a red square, and a circular icon with a gear. The output text is as follows:

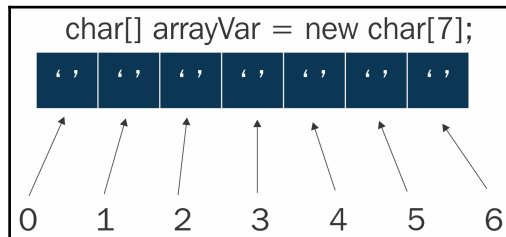
```
82  
83  
84  
85  
86  
87  
88  
89  
90  
91  
92  
93  
94  
95  
96  
97  
98  
99  
100  
BUILD SUCCESSFUL (total time: 0 seconds)
```

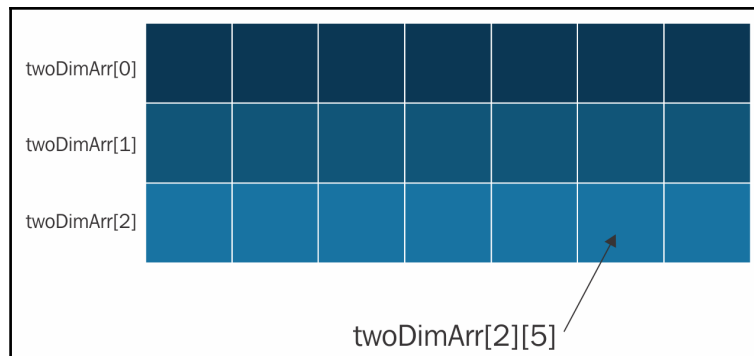
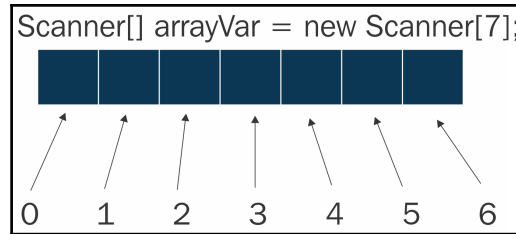
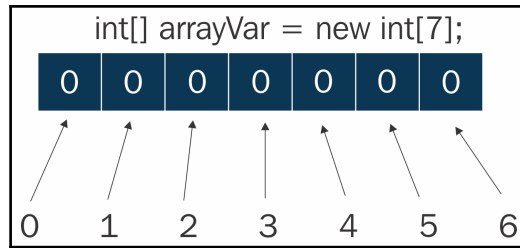
Chapter 4: Data Structures



Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040	 	Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	\$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	((72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051))	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[[123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174	|	
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135]]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177		DEL

Output - Alphabet (run) %	Variables	Breakpoints
 run:  [a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z]  BUILD SUCCESSFUL (total time: 0 seconds)		

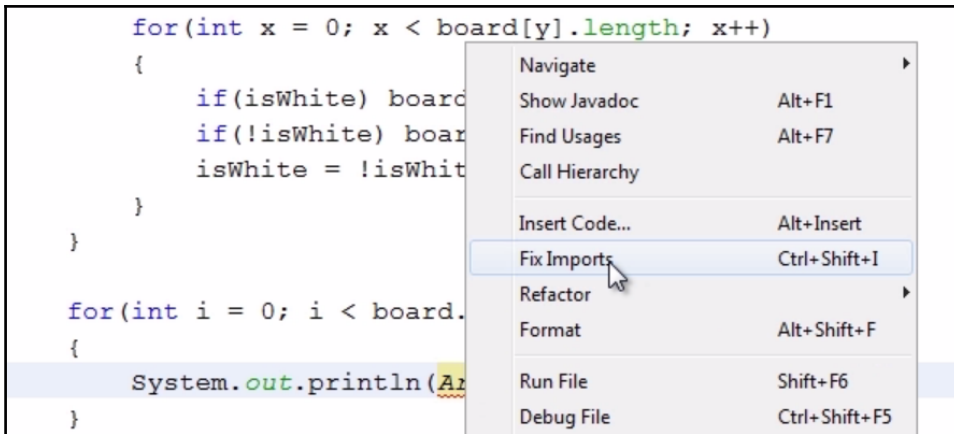




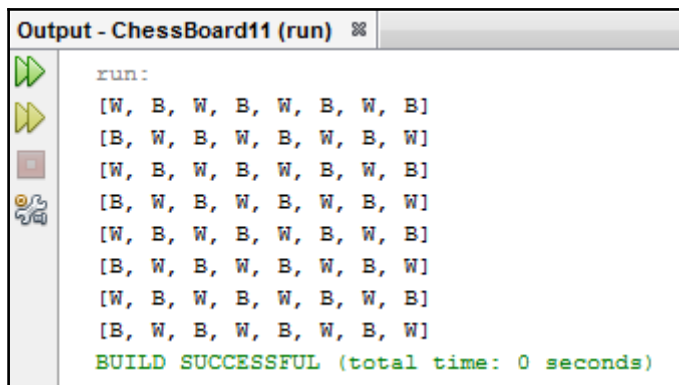
```
for(int i = 0; i < board.length; i++)  
{  
    System.out.println(Arrays.toString(board));  
}
```

```
for(int x = 0; x < board[y].length; x++)
{
    if(isWhite) board[x][y] = 'W';
    if(!isWhite) board[x][y] = 'B';
    isWhite = !isWhite;
}

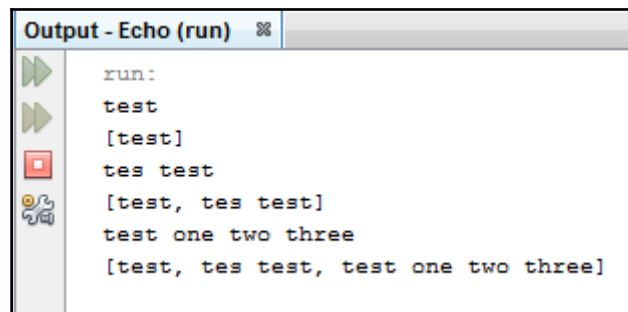
for(int i = 0; i < board.length; i++)
{
    System.out.println(ArrayUtils.toString(board[i]));
}
```



```
Output - ChessBoard11 (run) ✖
run:
[W, B, W, B, W, B, W, B]
[B, W, B, W, B, W, B, W]
[W, B, W, B, W, B, W, B]
[B, W, B, W, B, W, B, W]
[W, B, W, B, W, B, W, B]
[B, W, B, W, B, W, B, W]
[W, B, W, B, W, B, W, B]
[B, W, B, W, B, W, B, W]
BUILD SUCCESSFUL (total time: 0 seconds)
```



```
Output - Echo (run) ✖
run:
test
[test]
tes test
[test, tes test]
test one two three
[test, tes test, test one two three]
```



```
Output %
Echo (run) #4 % Echo (run) #5 %
run:
abc d efg
[abc d efg]
h ijkl m
[abc d efg, h ijkl m]
no pq
[abc d efg, h ijkl m, no pq]
rst uvwxyz
[abc d efg, h ijkl m, no pq, rst uvwxyz]
```

```
Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: -1
    at java.util.ArrayList.elementData(ArrayList.java:418)
    at java.util.ArrayList.get(ArrayList.java:431)
    at echo.Echo.main(Echo.java:18)
C:\Users\nidhishas\AppData\Local\NetBeans\Cache\8.2\executor-snippets\run.xml:53: Java returned: 1
```

```
Output - Echo (run) %
run:
abc def ghi
[abc def ghi ]
jkl mno pqr
[abc def ghi , jkl mno pqr]
stu vwx yz
[abc def ghi , jkl mno pqr, stu vwx yz]
CLEAR
[]
```

```
Output - Echo (run) %
run:
abc def ghi
[abc def ghi ]
jkl mno pqr
[abc def ghi , jkl mno pqr]
stu vwx yz
[abc def ghi , jkl mno pqr, stu vwx yz]
CLEAR
[]
END
BUILD SUCCESSFUL (total time: 2 minutes 35 seconds)
```

```
String[] allNames =
    //<editor-fold desc="raw names data">
    {"Jane", "Addams",
     "Muhammad", "Ali",
     "Stephen", "Ambrose",
     "Louis", "Armstrong",
     "Joan", "Baez",
     "Josephine", "Baker",
```

```
public static void main(String[] args) {
    String[] allNames =
        raw names data
    String[] firstNames = new String[allNames.length/2];
```

```
Output - Maps (run) %
run:
Jane
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
Output - Maps (run) %
run:
Frank
BUILD SUCCESSFUL (total time: 0 seconds)
```

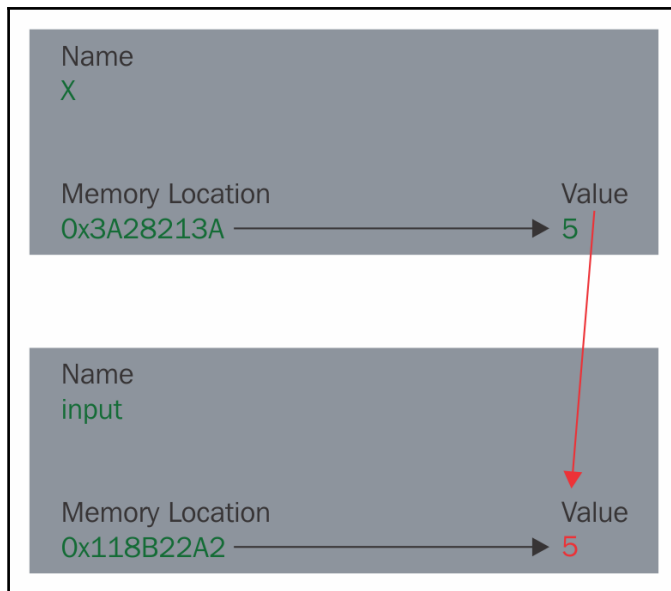
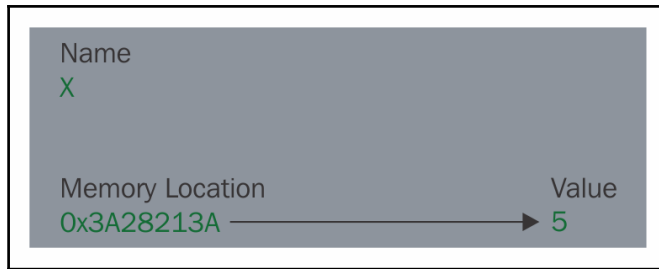
Chapter 5: Functions

```
Output - TemperatureConverter (run) #2 %  
run:  
Input type (F/C/K): F  
Output type (F/C/K): C  
Temperature: 32  
0.0  
BUILD SUCCESSFUL (total time: 8 seconds)
```

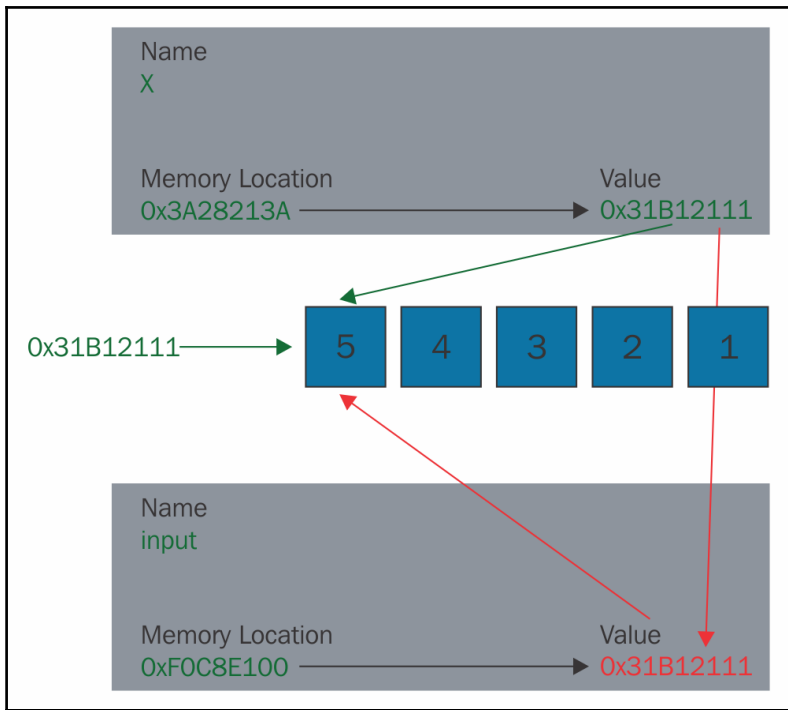
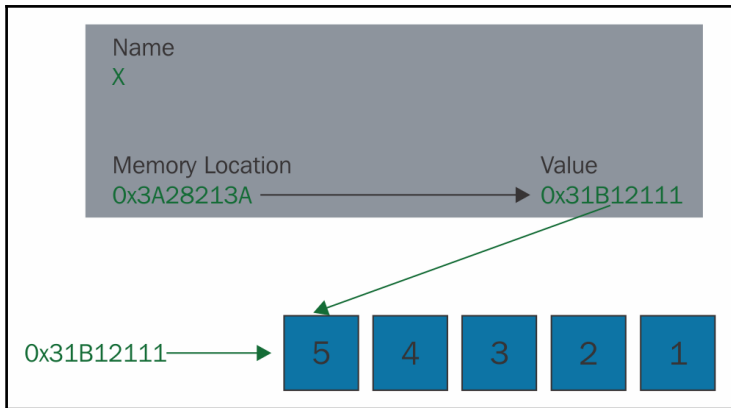
```
Output - TemperatureConverter (run) #2 %  
run:  
Input type (F/C/K): C  
Output type (F/C/K): C  
Temperature: 67.542452  
67.54245  
BUILD SUCCESSFUL (total time: 34 seconds)
```

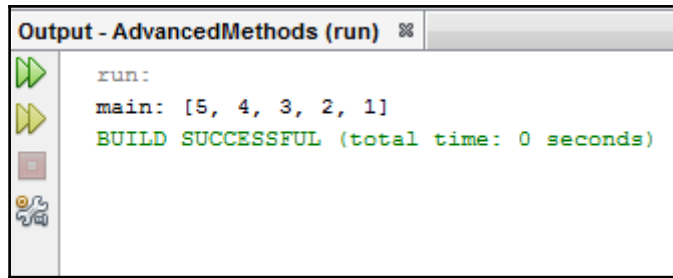
```
Output - TemperatureConverter (run) #2 %  
run:  
Input type (F/C/K): K  
Output type (F/C/K): F  
Temperature: 1  
-457.86996  
BUILD SUCCESSFUL (total time: 8 seconds)
```

```
Output - AdvancedMethods (run) %  
run:  
main: 5  
BUILD SUCCESSFUL (total time: 0 seconds)
```



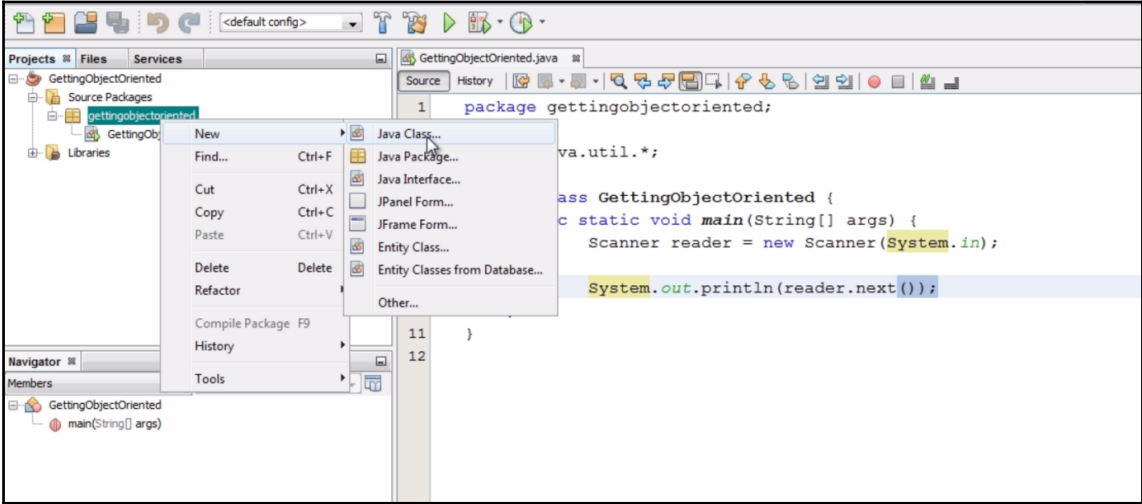
```
Output - AdvancedMethods (run) ✖
run:
main: [15, 14, 13, 12, 11]
BUILD SUCCESSFUL (total time: 0 seconds)
```

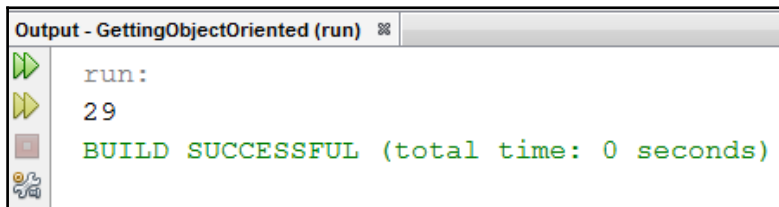
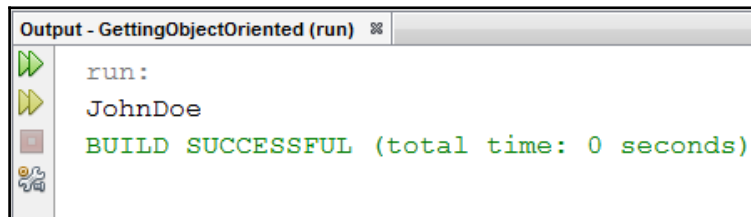
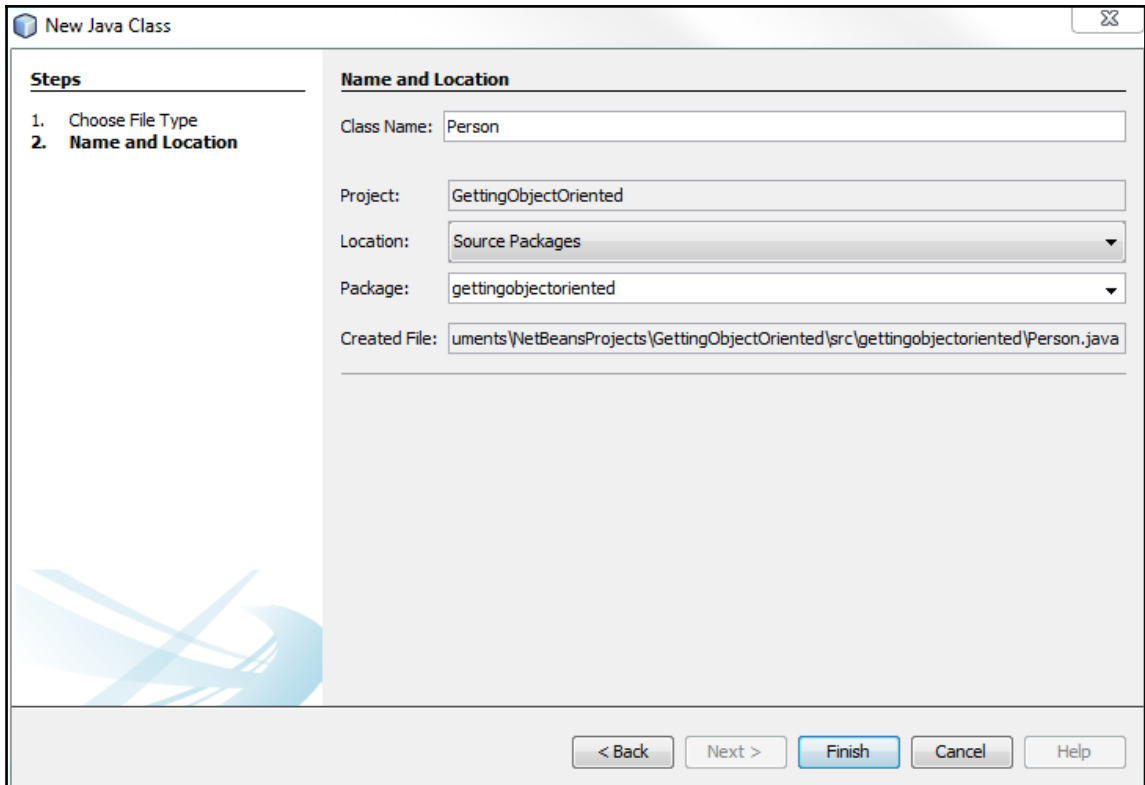




```
run:
main: [5, 4, 3, 2, 1]
BUILD SUCCESSFUL (total time: 0 seconds)
```


Chapter 6: Modeling with Object-Oriented Java





```
Output - GettingObjectOriented (run) ✖
run:
Exception in thread "main" java.lang.NullPointerException
    at gettingobjectoriented.Person.age(Person.java:13)
    at gettingobjectoriented.GettingObjectOriented.main(GettingObjectOriented.java:15)
C:\Users\nidhishas\AppData\Local\NetBeans\Cache\8.2\executor-snippets\run.xml:53: Java returned: 1
BUILD FAILED (total time: 0 seconds)
```

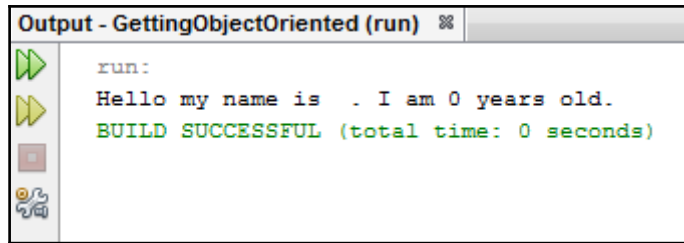
```
import java.util.*;
public class GettingObjectOriented {
    public static void main(String[] args) {
        Person john = new Person();
    }
}
```

constructor Person in class Person cannot be applied to given types;
required: String,String,Calendar
found: no arguments
reason: actual and formal argument lists differ in length
(Alt-Enter shows hints)

```
public Person(String firstName, String lastName, Calendar birthday)
{
    this.firstName = firstName;
    this.lastName = lastName;
    this.birthday = birthday;
}
```

```
Output - GettingObjectOriented (run) ✖
run:
Hello my name is John Doe. I am 29 years old.
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
Output - GettingObjectOriented (run) ✖
run:
Hello my name is John Doe. I am 0 years old.
BUILD SUCCESSFUL (total time: 0 seconds)
```



```
run:
Hello my name is . I am 0 years old.
BUILD SUCCESSFUL (total time: 0 seconds)
```

Chapter 7: More Object-Oriented Java

```
Inheritance.java Book.java
Source History
2 public class Book {
3     private String title;
4     private String author;
5     private String publisher;
6     private String genre;
7
8     public Book(String title, String author, String publisher, String genre)
9     {
10        this.title = title;
11        this.author = author;
12        this.publisher = publisher;
13        this.genre = genre;
14    }
15
16    public void Print()
17    {
18        System.out.println(title);
19        System.out.println("\tWritten By: " + author);
20        System.out.println("\tPublished By: " + publisher);
21        System.out.println("\tIs A: " + genre);
22    }
23 }

Poem.java
Source History
1 package inheritance;
2 public class Poem {
3     private String title;
4     private String author;
5     private String style;
6
7     public Poem(String title, String author, String style)
8     {
9        this.title = title;
10        this.author = author;
11        this.style = style;
12    }
13
14    public void Print()
15    {
16        System.out.println(title);
17        System.out.println("\tWritten By: " + author);
18        System.out.println("\tIn The Style Of: " + style);
19    }
20 }
```

```
Inheritance.java Book.java
Source History
16 public void Print()
17 {
18     System.out.println(title);
19     System.out.println("\tWritten By: " + author);
20     System.out.println("\tPublished By: " + publisher);
21     System.out.println("\tIs A: " + genre);
22 }
23 }

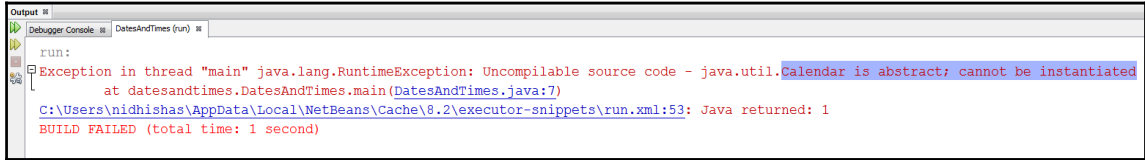
Poem.java
Source History
14 public void Print()
15 {
16     System.out.println(title);
17     System.out.println("\tWritten By: " + author);
18     System.out.println("\tIn The Style Of: " + style);
19 }
20 }
21 }
```

```
Output - Inheritance (run)
run:
The Lord Of The Rings
    Written By: J.R.R. Tolkein
    Published By: George Allen and Unwin
    Is A: Fantasy
The Iliad
    Written By: Homer
    In The Style Of: Dactylic Hexameter
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
Output - Inheritance (run) %
run:
The Lord Of The Rings
    Written By: J.R.R. Tolkein
    Published By: George Allen and Unwin
    Is A: Fantasy
The Iliad
    Written By: Homer
    In The Style Of: Dactylic Hexameter
BUILD SUCCESSFUL (total time: 0 seconds)
```

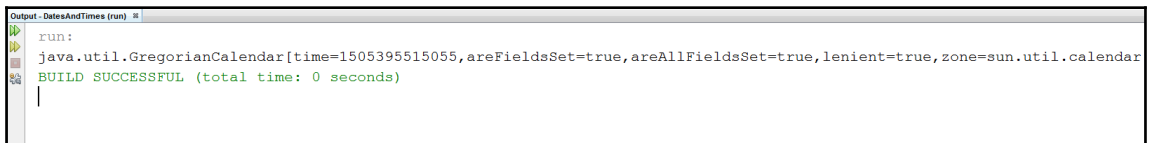
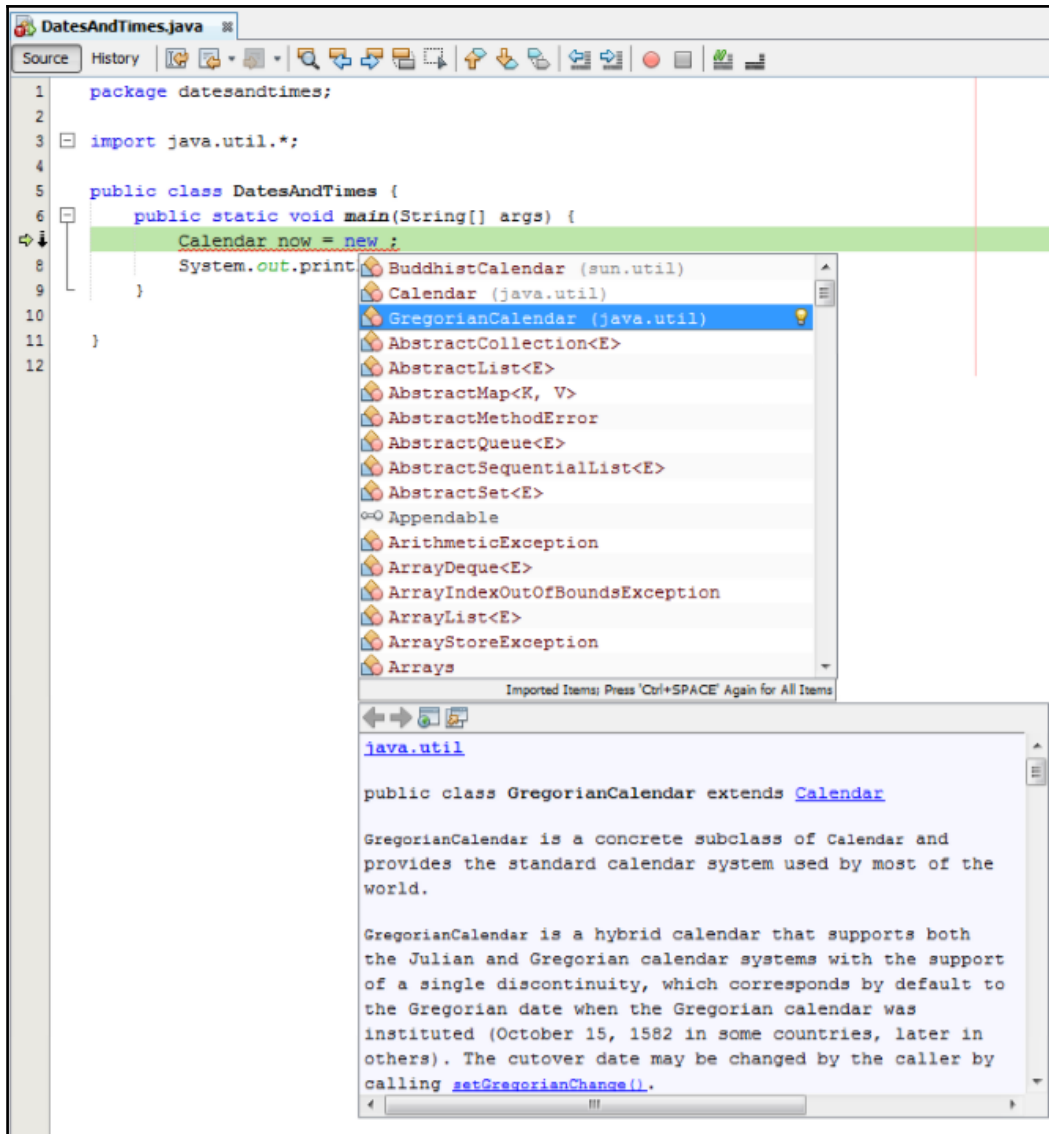
```
Output %
Debugger Console % Inheritance (run) %
    Published By: George Allen and Unwin
    Is A: Fantasy
POEM: The Iliad
    Written By: Homer
    In The Style Of: Dactylic Hexameter
Java
    Written By: Zach
POEM: The Iliad
    Written By: Homer
    In The Style Of: Dactylic Hexameter
The Lord Of The Rings
    Written By: J.R.R. Tolkein
    Published By: George Allen and Unwin
    Is A: Fantasy
BUILD SUCCESSFUL (total time: 0 seconds)
```

Chapter 8: Useful Java Classes



The screenshot shows an IDE's Output window with the following text:

```
run:  
Exception in thread "main" java.lang.RuntimeException: Uncompilable source code - java.util.Calendar is abstract; cannot be instantiated  
    at datesandtimes.DatesAndTimes.main(DatesAndTimes.java:7)  
C:\Users\nidhishas\AppData\Local\NetBeans\Cache\8.2\executor-snippets\run.xml:53: Java returned: 1  
BUILD FAILED (total time: 1 second)
```

The screenshot shows an IDE window with the following content:

```

1 package datesandtimes;
2
3 import java.util.*;
4
5 public class DatesAndTimes {
6     public static void main(String[] args) {
7         Calendar now = new GregorianCalendar();
8         System.out.println(now.);
9         System.out.println(n
10        System.out.println(n
11    }
12
13 }
14

```

The class browser on the right lists the following methods:

- after(Object when) boolean
- before(Object when) boolean
- clone() Object
- compareTo(Calendar anotherCalendar) int
- equals(Object obj) boolean
- get(int field) int**
- getActualMaximum(int field) int
- getActualMinimum(int field) int
- getCalendarType() String
- getClass() Class<?>
- getDisplayName(int field, int style, Locale locale) String
- getDisplayNames(int field, int style, Locale locale) Map<String, Integer>
- getFirstDayOfWeek() int
- getGreatestMinimum(int field) int
- getLeastMaximum(int field) int
- getMaximum(int field) int
- getMinimalDaysInFirstWeek() int

The detail pane for the selected method `java.util.Calendar.get(int field)` shows the following information:

```

public int get(int field)

Returns the value of the given calendar field. In lenient
mode, all calendar fields are normalized. In non-lenient
mode, all calendar fields are validated and this method
throws an exception if any calendar fields have out-of-
range values. The normalization and validation are handled
by the Calendar.complete\(\) method, which process is calendar
system dependent.

Parameters:
    field - the given calendar field.

Returns:

```

The screenshot shows an IDE with a source code editor on the left and a documentation window on the right. The source code is as follows:

```

1 package datesandtimes;
2
3 import java.util.*;
4
5 public class DatesAndTimes {
6     public static void main(String[] args) {
7         Calendar now = new GregorianCalendar();
8         System.out.println(now.get(Calendar.YEAR));
9         System.out.println(now);
10        System.out.println(now);
11    }
12
13 }
14

```

The documentation window displays the following information for `java.util.Calendar`:

```

java.util.Calendar

public static final int ALL_STYLES = 0

A style specifier for getDisplayNames indicating names in all
styles, such as "January" and "Jan".

Since:
    1.6

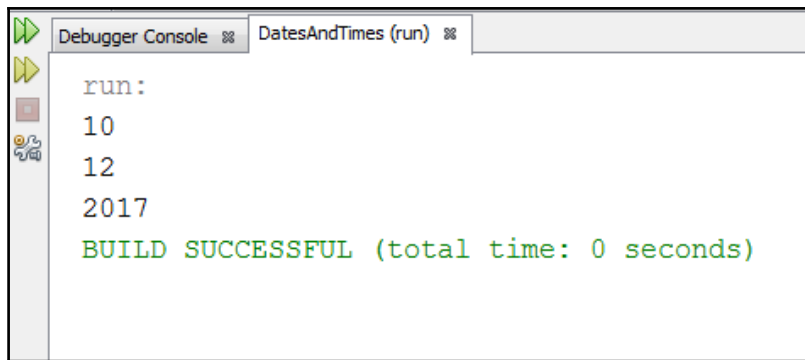
See Also:
    Calendar.SHORT\_FORMAT, Calendar.LONG\_FORMAT, Calendar.SHORT\_STANDALONE, Calendar.LONG\_STANDALONE, Calendar.SHORT, Calendar.LONG

```

```

Output - DatesAndTimes (run) %
run:
9
12
2017
BUILD SUCCESSFUL (total time: 0 seconds)

```



The image shows a screenshot of a debugger console window. The window has a title bar with two tabs: "Debugger Console" and "DatesAndTimes (run)". On the left side of the console, there are several icons: a green play button, a yellow play button, a red square, and a magnifying glass. The main area of the console contains the following text:

```
run:  
10  
12  
2017  
BUILD SUCCESSFUL (total time: 0 seconds)
```

```

1 package datesandtimes;
2
3 import java.util.*;
4
5 public class DatesAndTimes {
6     public static void main(String[] args) {
7         Calendar now = new GregorianCalendar();
8
9         now.
10
11     }
12
13 }
14
15
16

```

Instance Members: Press 'Ctrl+SPACE' Again for All Items

[java.util.Calendar](#)

public long `getTimeInMillis()`

Returns this Calendar's time value in milliseconds.

Returns:

the current time as UTC milliseconds from the epoch.

See Also:

[Calendar.getTime\(\)](#), [Calendar.setTimeInMillis\(long\)](#)

Output - DatesAndTimes (run) ✖

```

run:
1507801219947

```

```
Output - DatesAndTimes (run) ⌘
run:
0
1
1
1970
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
Output - Strings (run) ⌘
run:
string1: Strings are arrays of characters
string2: Strings are arrays of characters
false
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
Output - Strings (run) ⌘
run:
string1: Strings are arrays of characters
string2: Strings are arrays of characters
true
BUILD SUCCESSFUL (total time: 0 seconds)
```

```

1 package strings;
2
3 public class Strings {
4
5     public static void main(String[] args) {
6         String s1 = new String("Strings are arrays of characters");
7         String s2 = new String("Strings are arrays of characters");
8
9         System.out.println("string1: " + s1.);
10        System.out.println("string2: " +
11        System.out.println(s1.equals(s2))
12    }
13 }
14

```

Method list for `String`:

- `concat(String str)` String
- `intern()` String
- `replace(CharSequence target, CharSequence replacement)` String
- `replace(char oldChar, char newChar)` String
- `replaceAll(String regex, String replacement)` String
- `replaceFirst(String regex, String replacement)` String
- `substring(int beginIndex)` String
- `substring(int beginIndex, int endIndex)` String
- `toLowerCase()` String
- `toLowerCase(Locale locale)` String
- `toString()` String
- `toUpperCase()` String
- `toUpperCase(Locale locale)` String
- `trim()` String
- `charAt(int index)` char**
- `chars()` IntStream
- `codePointAt(int index)` int

Instance Members: Press 'Ctrl+SPACE' Again for All Items

`java.lang.String`

```

public char charAt(int index)

Returns the char value at the specified index. An index
ranges from 0 to length() - 1. The first char value of the
sequence is at index 0, the next at index 1, and so on, as
for array indexing.

If the char value specified by the index is a surrogate,
the surrogate value is returned.

Parameters:
    index - the index of the char value.

Returns:

```

```

Output - Strings (run)
run:
string1: Strings are arrays of char
string2: Strings are arrays of characters
true
BUILD SUCCESSFUL (total time: 0 seconds)

```

```

Output - Strings (run)
run:
>> string1: Strings are arrays of char <<
>> string2: Strings are arrays of characters <<
BUILD SUCCESSFUL (total time: 0 seconds)

```

```
Output - Exceptions (run) ❏
run:
Input a number: 542.64
You input the number: 542.64

Input a number: 56425
You input the number: 56425.0

Input a number:
```

```
Output - Exceptions (run) #2 ❏
run:
Input a number: fail
Exception in thread "main" java.util.InputMismatchException
    at java.util.Scanner.throwFor(Scanner.java:864)
    at java.util.Scanner.next(Scanner.java:1485)
    at java.util.Scanner.nextFloat(Scanner.java:2345)
    at exceptions.Exceptions.main(Exceptions.java:11)
C:\Users\nidhishas\AppData\Local\NetBeans\Cache\8.2\executor-snippets\run.xml:53: Java returned: 1
BUILD FAILED (total time: 5 seconds)
```

```
Output - Exceptions (run) ❏
Input a number: Please enter a float number.

Input a number: Please enter a float number.

Input a number: Please enter a float number.

Input a number: Please enter a float number.
```



```
Output - Exceptions (run) #3
run:
Input a number: 5432.3
You input the number: 5432.3

Input a number: next
You passed inavlid input. Not a flaot!

Input a number:
```

```
Output - Exceptions (run) #6
run:
Input a number: fail
You passed inavlid input. Not a flaot!
java.util.InputMismatchException
    at java.util.Scanner.throwFor(Scanner.java:864)
    at java.util.Scanner.next(Scanner.java:1485)
    at java.util.Scanner.nextFloat(Scanner.java:2345)
    at exceptions.Exceptions.main(Exceptions.java:12)
```

```
Output - TheObjectClass (run)
run:
A MyClass object was created with value:abcdefg
A MyClass object was created with value:abcdefg
MyMethod was called on a MyClass object with value: abcdefg
MyMethod was called on a MyClass object with value: abcdefg
The objects are the same: false
The objects are the same: false
BUILD SUCCESSFUL (total time: 0 seconds)
```

Modifier and Type	Method and Description
protected Object	clone() Creates and returns a copy of this object.
boolean	equals(Object obj) Indicates whether some other object is "equal to" this one.
protected void	finalize() Called by the garbage collector on an object when garbage collection determines that there are no more references to the object.
Class<?>	getClass() Returns the runtime class of this Object.
int	hashCode() Returns a hash code value for the object.
void	notify() Wakes up a single thread that is waiting on this object's monitor.
void	notifyAll() Wakes up all threads that are waiting on this object's monitor.
String	toString() Returns a string representation of the object.
void	wait() Causes the current thread to wait until another thread invokes the notify() method or the notifyAll() method for this object.
void	wait(long timeout) Causes the current thread to wait until either another thread invokes the notify() method or the notifyAll() method for this object, or a specified amount of time has elapsed.
void	wait(long timeout, int nanos) Causes the current thread to wait until another thread invokes the notify() method or the notifyAll() method for this object, or some other thread interrupts the current thread, or a certain amount of real time has elapsed.

```

Output - TheObjectClass (run)
run:
A MyClass object was created with value:abcdefg
A MyClass object was created with value:abcdefg
MyMethod was called on a MyClass object with value: abcdefg
MyMethod was called on a MyClass object with value: abcdefg
The objects are the same: false
The objects are the same: true
BUILD SUCCESSFUL (total time: 0 seconds)

```

```

1 package the.primitiveclasses;
2
3 public class ThePrimitiveClasses {
4
5     public static void main(String[] args) {
6         String s = "string";
7
8
9         Character c = 'C';
10        c.
11    }
12
13 }
14

```

charValue()	char
compareTo(Character anotherCharacter)	int
equals(Object obj)	boolean
getClass()	Class<?>
hashCode()	int
notify()	void
notifyAll()	void
toString()	String
wait()	void
wait(long timeout)	void
wait(long timeout, int nanos)	void

Instance Members; Press 'Ctrl+SPACE' Again for All Items

```

< >
java.lang.Character

public int compareTo(Character anotherCharacter)

Compares two Character objects numerically.

Parameters:
    anotherCharacter - the Character to be compared.

Returns:
    the value 0 if the argument Character is equal to
    this Character; a value less than 0 if this
    Character is numerically less than the Character
    argument; and a value greater than 0 if this
    Character is numerically greater than the Character
    argument (unsigned comparison). Note that this is

```

Output - The Primitiv

```

run:
false
BUILD S

```

```

1 package the.primitiveclasses;
2
3 public class ThePrimitiveClasses {
4
5     public static void main(String[] args) {
6         String s = "string";
7
8
9         Character c = 'C';
10        System.out.println(Character.isl());
11    }
12
13 }
14

```

isLetter(char ch)	boolean
isLetter(int codePoint)	boolean
isLetterOrDigit(char ch)	boolean
isLetterOrDigit(int codePoint)	boolean
isLowSurrogate(char ch)	boolean
isLowerCase(char ch)	boolean
isLowerCase(int codePoint)	boolean

```

Output - The PrimitiveClasses (run)
run:
false
BUILD SUCCESSFUL (total time:

```

[java.lang.Character](#)

public static boolean isLowerCase(char ch)

Determines if the specified character is a lowercase character.

A character is lowercase if its general category type, provided by `Character.getType(ch)`, is `LOWERCASE_LETTER`, or it has contributory property `Other_Lowercase` as defined by the Unicode Standard.

The following are examples of lowercase characters:

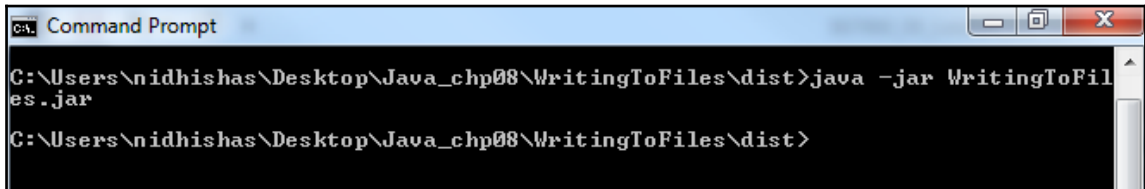
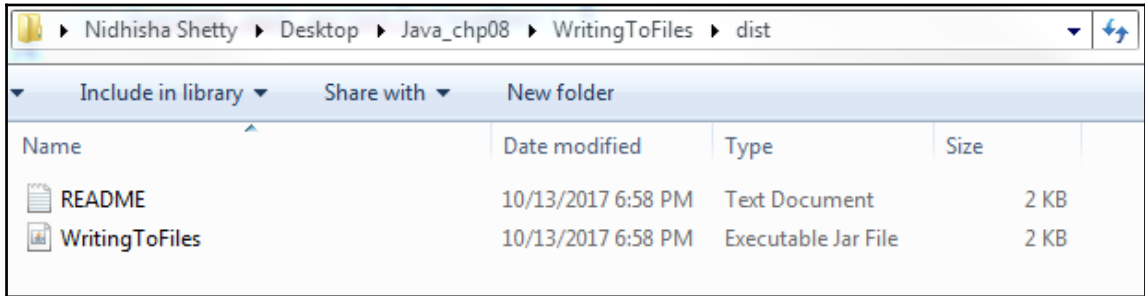
a b c d e f g h i j k l m n o p q r s t u v w x y z
 11-0000FE1 11-0000FE2 11-0000FE3 11-0000FE4 11-0000FE5 11-0000FE6

```

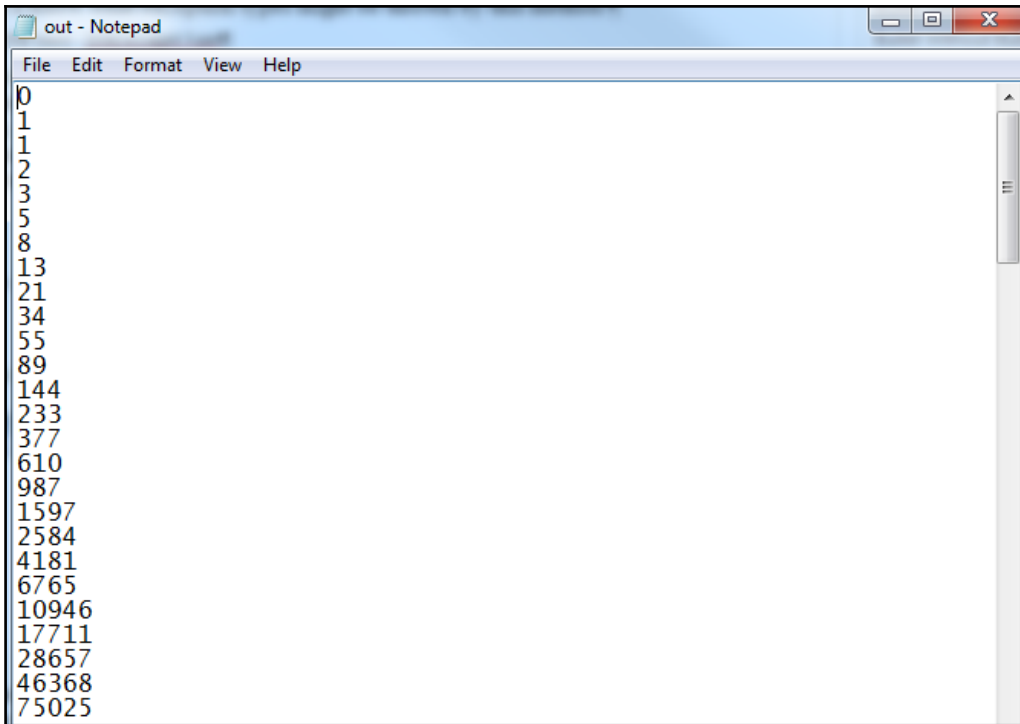
Output - The PrimitiveClasses (run)
run:
true
BUILD SUCCESSFUL (total time: 0 seconds)

```

Chapter 9: File Input and Output



```
catch (IOEx unreported exception IOException; must be caught or declared to be thrown
    System Dereferencing possible null pointer
}
finally{
    out.close();
}
```



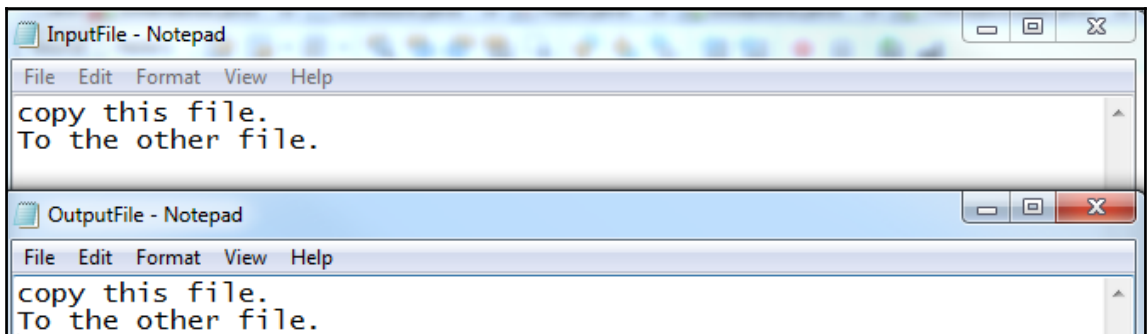
out - Notepad

```
File Edit Format View Help
0
1
1
2
3
5
8
13
21
34
55
89
144
233
377
610
987
1597
2584
4181
6765
10946
17711
28657
46368
75025
```



OutputFile - Notepad

```
File Edit Format View Help
c
```



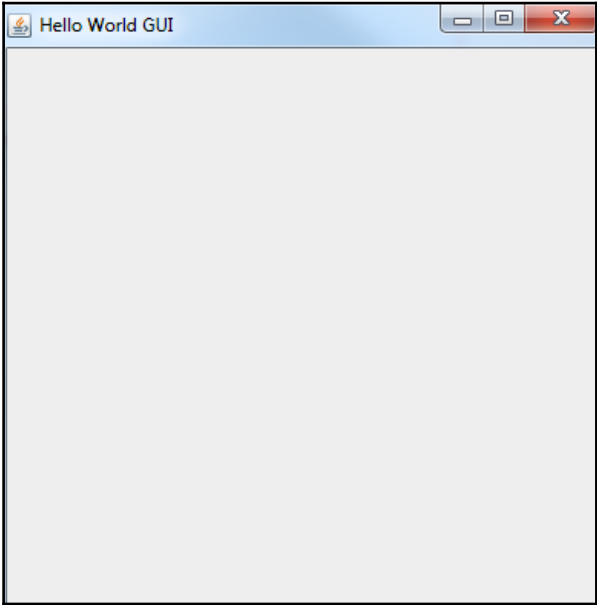
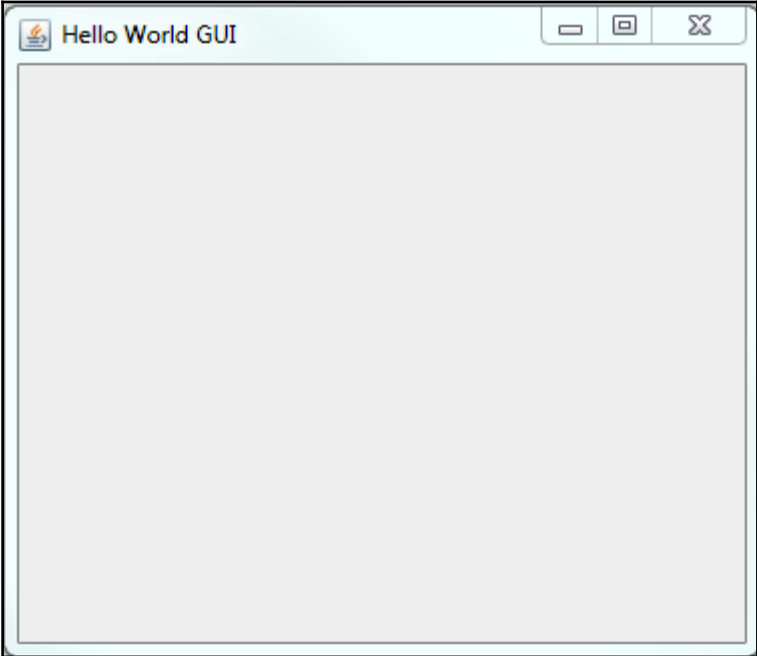
InputFile - Notepad

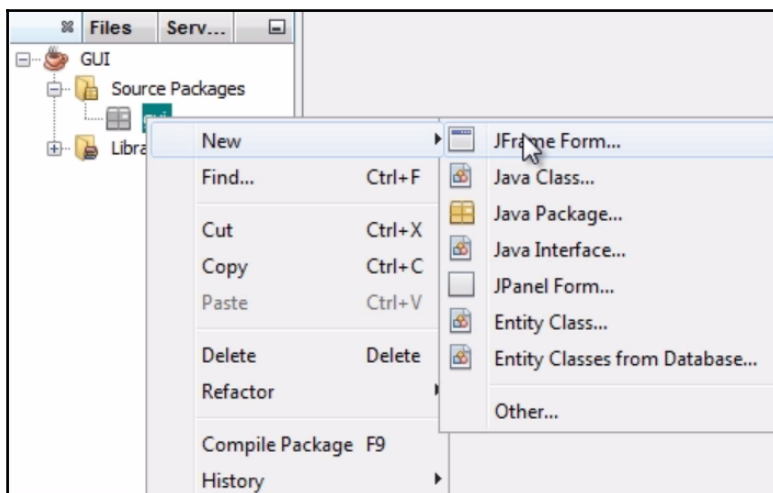
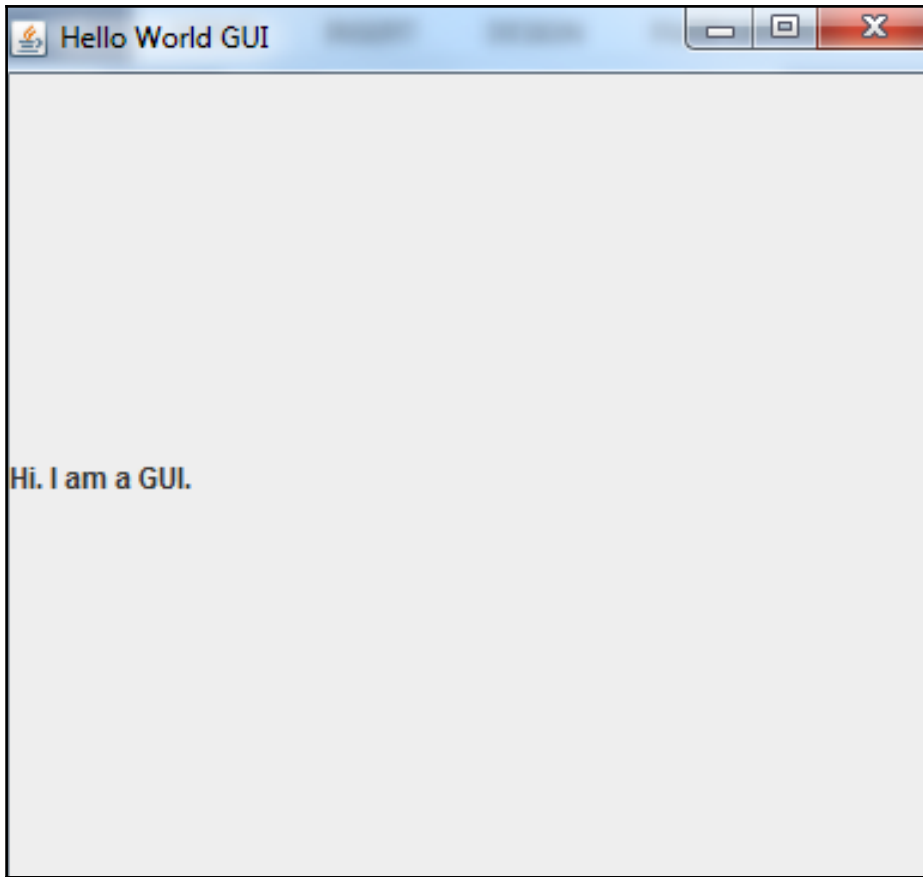
```
File Edit Format View Help
copy this file.
To the other file.
```

OutputFile - Notepad

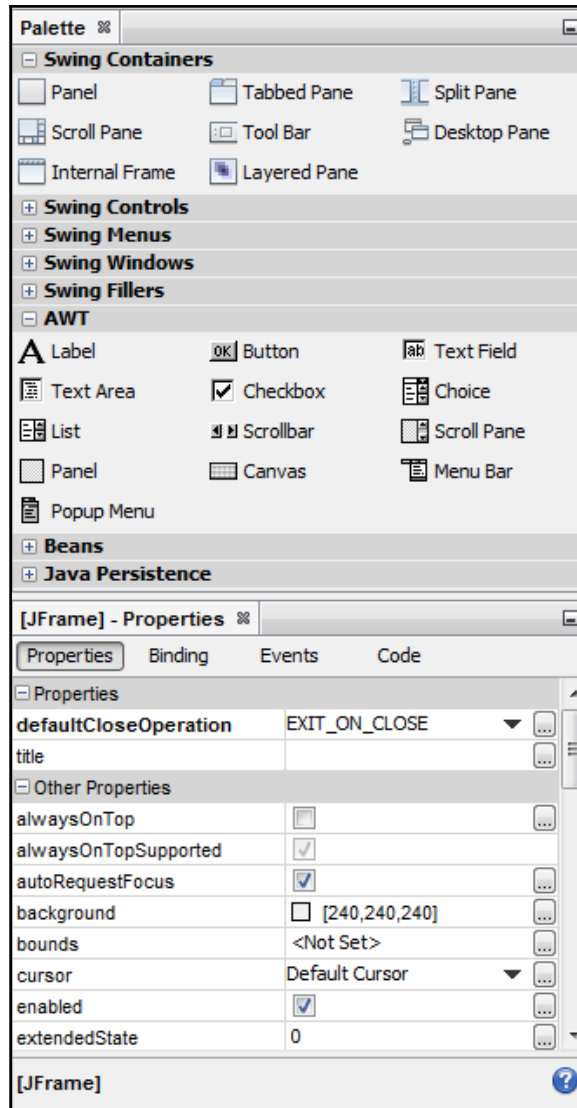
```
File Edit Format View Help
copy this file.
To the other file.
```

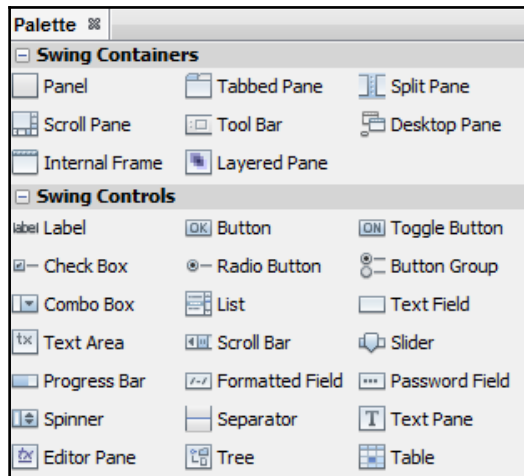
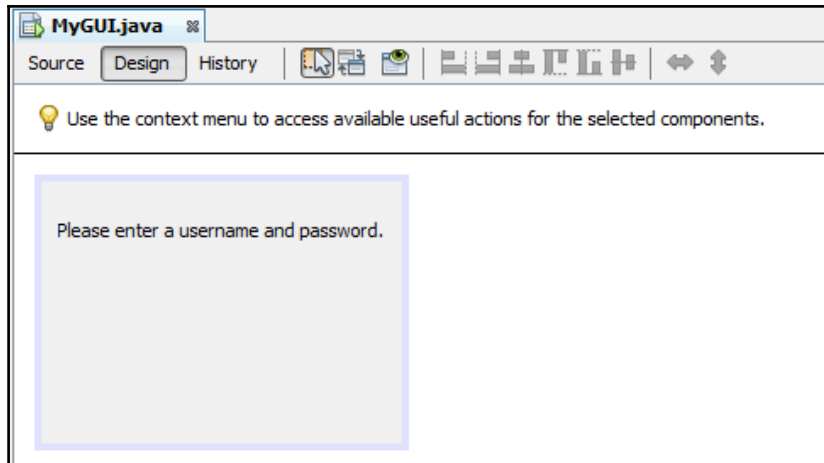

Chapter 10: Basic GUI Development

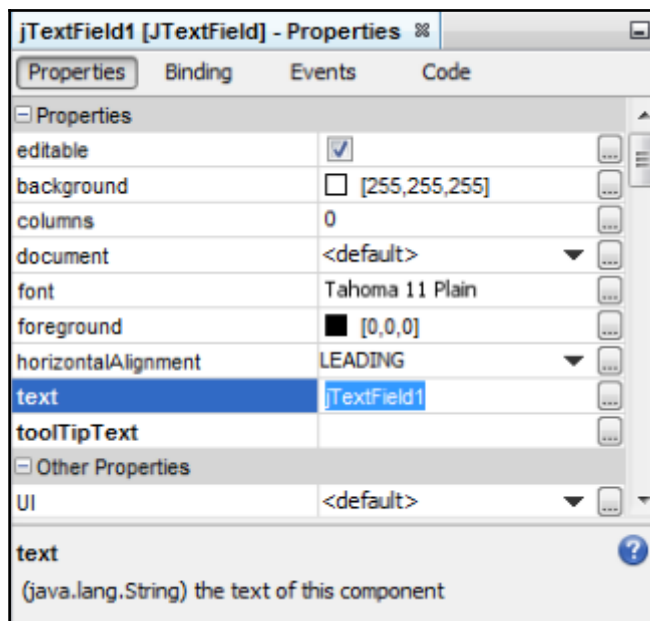
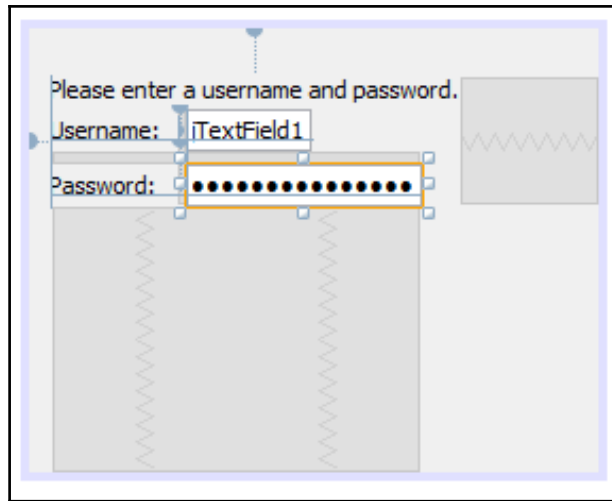


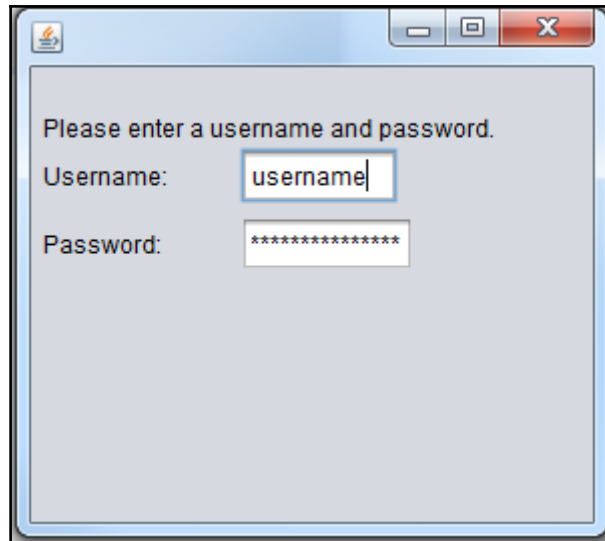


```
1  /*
2  * To change this license header, choose License Headers in Project Properties.
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6  package gui;
7
8  /**
9  *
10 * @author nidhishas
11 */
12 public class MyGUI extends javax.swing.JPanel {
13
14     /**
15     * Creates new form MyGUI
16     */
17     public MyGUI() {
18         initComponents();
19     }
20
21     /**
22     * This method is called from within the constructor to initialize the form.
23     * WARNING: Do NOT modify this code. The content of this method is always
24     * regenerated by the Form Editor.
25     */
26     @SuppressWarnings("unchecked")
```

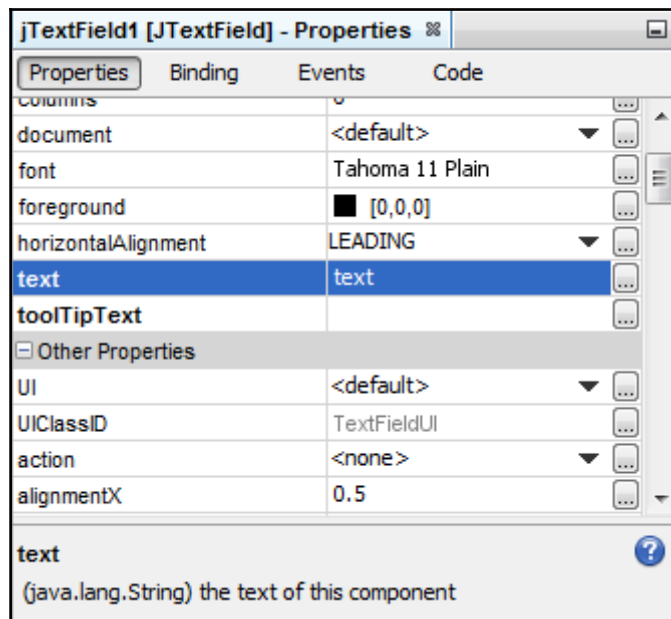
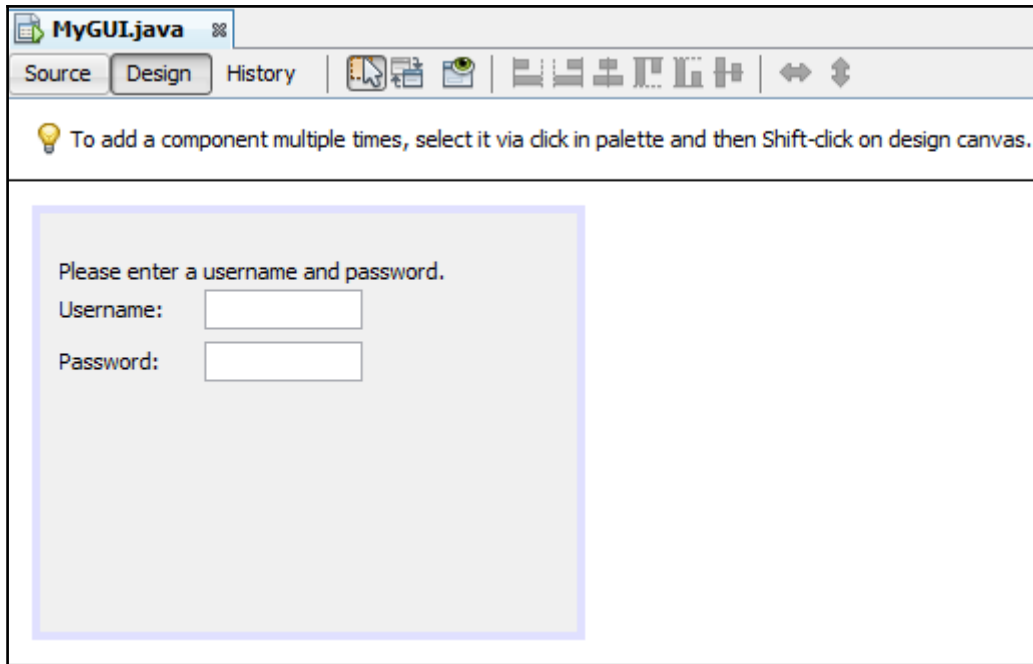








```
public static void main(String args[]) {  
    /* Set the Nimbus look and feel */  
    Look and feel setting code (optional)  
  
    /* Create and display the form */  
    java.awt.EventQueue.invokeLater(new Runnable() {  
        public void run() {  
            new MyGUI().setVisible(true);  
        }  
    });  
};
```



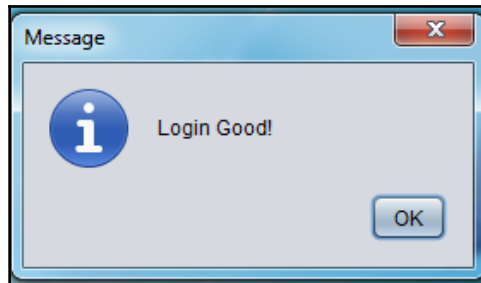


A screenshot of a login form window. The window has a blue title bar with a small icon on the left and standard minimize, maximize, and close buttons on the right. The main content area is light gray and contains the following text and controls:

Please enter a username and password.

Username:

Password:



Chapter 11: XML

A screenshot of an IDE's output window titled "Output - LoadingAnXMLFile (run)". The window contains the following XML code:

```
run:
<?xml version="1.0" encoding="UTF-8" standalone="no"?><cars>
  <owner name="Billy">
    <car vin="LJPCBLCX11000237">
      <make>Ford</make>
      <model>Fusion</model>
      <year>2014</year>
      <color>Blue</color>
    </car>
    <car vin="LGHIALCX89880011">
      <make>Toyota</make>
      <model>Tacoma</model>
      <year>2013</year>
      <color>Green</color>
    </car>
    <car vin="GJSIALSS22000567">
      <make>Dodge</make>
```

```

10 public class ReadingXML {
11     public static void main(String[] args) {
12         Document dom;
13         DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
14         try {
15             DocumentBuilder docBuilder = factory.newDocumentBuilder();
16             dom = docBuilder.parse("cars.xml");
17
18             // Now, print out all of Jane's cars...
19             dom.getd
20         } catch (getDoctype()      DocumentType
21             catch (getDocumentElement() Element) {
22             S: getDocumentURI()      String);
23         } catch (getDomConfig() DOMConfiguration
24         catch (← → [Icons]
25             S: org.w3c.dom.Document
26         }
27         catch (public Element getDocumentElement()
28             S:
29         }
30     }
31 }
32

```

This is a convenience attribute that allows direct access to the child node that is the document element of the document.

```

Output - ReadingXML (run) ✖
run:
2000 Green Nissan Altima, vin:LLOKAJSS55548563
2013 Red Dodge Challenger, vin:OOKINAFS98111001
BUILD SUCCESSFUL (total time: 0 seconds)

```

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?><cars>
  <owner name="Mike">
    <car vin="LJPCBLCX11000237">
      <make>Ford</make>
      <model>Fusion</model>
      <year>2014</year>
      <color>Blue</color>
    </car>
    <car vin="LGHIALCX89880011">
      <make>Toyota</make>
      <model>Tacoma</model>
      <year>2013</year>
      <color>Green</color>
    </car>
    <car vin="GJSIALSS22000567">
      <make>Dodge</make>
      <model>Charger</model>
      <year>2013</year>
      <color>Red</color>
    </car>
  </owner>
  <owner name="Mike">
```