



Main.java

```
1- public class TypeCasting{
2  public static void main(String[] args)
3- {
4  //Implicit casting (automatic type conversion)
5  int myInt=9;
6  double myDouble = myInt;
7
8
9  System.out.println("Int value: "+myInt);
10 System.out.println("Converted to double: "+myDouble);
11
12
13 //Explicit casting (manual type conversion)
14 double anotherDouble=9.78;
15 int anotherInt = (int) anotherDouble;
16
17
18 System.out.println("Double value:"+anotherDouble);
19 System.out.println("Converted to int: "+anotherInt);
20 }
21 }
22
```

Run

Output


```
java -cp /tmp/nzyiRKi0v1/TypeCasting
Int value: 9
Converted to double: 9.0
Double value:9.78
Converted to int: 9
```

=== Code Execution Successful ===



Programiz
Online Java Compiler

Programiz PRO

Premium Coding Courses
by Programiz

[Learn More](#)

Main.java



Run

Output

java -cp /tmp/W5nviNVqhC/Main
Model: Toyota CorollaYear: 2020

=== Code Execution Successful ===

```
1 class Car {  
2     private String model;  
3     private int year;  
4     // Constructor  
5     public Car(String model, int year){  
6         this.model =model;  
7         this.year=year;  
8     }  
9     public void displayInfo() {  
10        System.out.println("Model: "+model+"Year: "+year);  
11    }  
12 }  
13 public class Main {  
14     public static void main(String[] args) {  
15         Car car1 = new Car("Toyota Corolla", 2020);  
16         car1.displayInfo();  
17     }  
18 }
```

2024/05/02 15:48

Programiz
Online Java Compiler

Programiz PRO Premium Coding Courses by Programiz [Learn More](#)

Main.java

```
1 class AccessSpecifierDemo {
2     public int publicVar =100; //Accessible from any other class
3     private int privateVar=200; // Accessible only within the class
4     protected int protectedVar =300; // Accessible within the class and by derived
      classes
5     public void display() {
6         System.out.println("Public: " + publicVar);
7         System.out.println("Private: " + privateVar);
8         System.out.println("Protected: "+protectedVar);
9     }
10 }
11 public class Main {
12     public static void main(String[] args) {
13
14         AccessSpecifierDemo demo=new AccessSpecifierDemo();
15         demo.display();
16         System.out.println(demo.publicVar);
17         // System.out.println(demo.privateVar); // Error: Cannot access
18         // System.out.println(demo.protectedVar); // Error: Cannot access outside the
           class without inheritance
19     }
20 }
```

Run

Output

```
java -cp /tmp/ttoyPR9gxC/Main
Public:100
Private: 200
Protected: 300
100
=== Code Execution Successful ===
```



Main.java

```
1 public class Car {
2     // Fields (attributes)
3     private String color;
4     private String model;
5     // Constructor
6     public Car(String color, String model) {
7         this.color=color;
8         this.model=model;
9     }
10    //Method
11    public void display() {
12        System.out.println("Car model: "+model+", Color: "+color);
13    }
14    // Main method to create and use objects of Car class
15    public static void main(String[] args)
16    {
17        Car myCar = new Car("Red", "Toyota Corolla");
18        myCar.display();
19    }
20 }
21
```



Run

Output

```
java -cp /tmp/thwFqQCDxD/Car
Car model: Toyota Corolla, Color: Red
```

```
=== Code Execution Successful ===
```

2024/05/04 20:01

Programiz
Online Java Compiler

Programiz PRO Premium Coding Courses by Programiz [Learn More](#)

Main.java

```
1 class Animal {
2     public void sound() {
3         System.out.println("Some sound");
4     }
5 }
6 class Dog extends Animal {
7     @Override
8     public void sound() {
9         System.out.println("Woof");
10    }
11 }
12 class Cat extends Animal {
13     @Override
14     public void sound() {
15         System.out.println("Meow");
16     }
17 }
18 public class Main {
19     public static void main(String[] args) {
20         Animal myAnimal = new Animal();
21         Animal myDog = new Dog();
22         Animal myCat = new Cat();
23         myAnimal.sound(); // Outputs Some sound
24     }
25 }
26
```

Run

Output

```
java -cp /tmp/2zjvqqiN0w/Main
Some sound
=== Code Execution Successful ===
```

BSE midcap +1.39%

ENG IN 15:19 02-05-2024