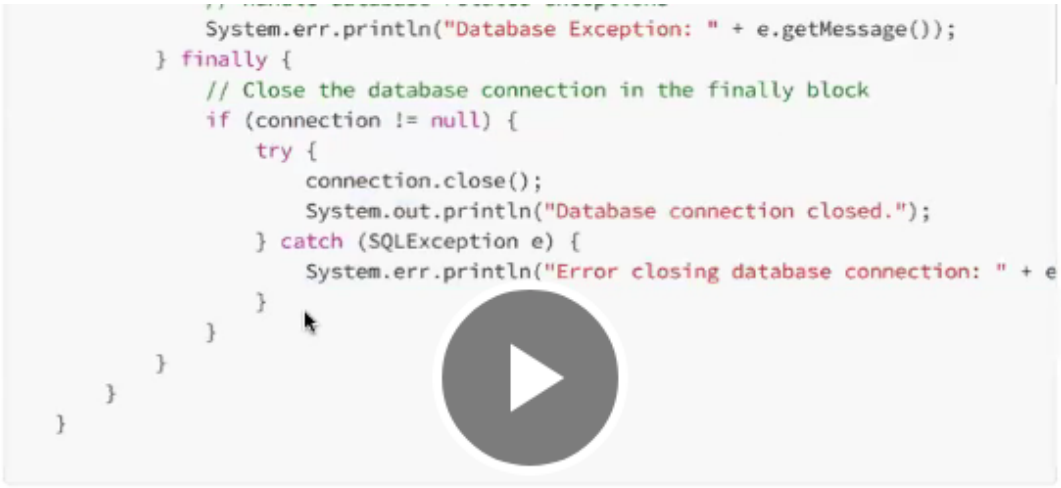
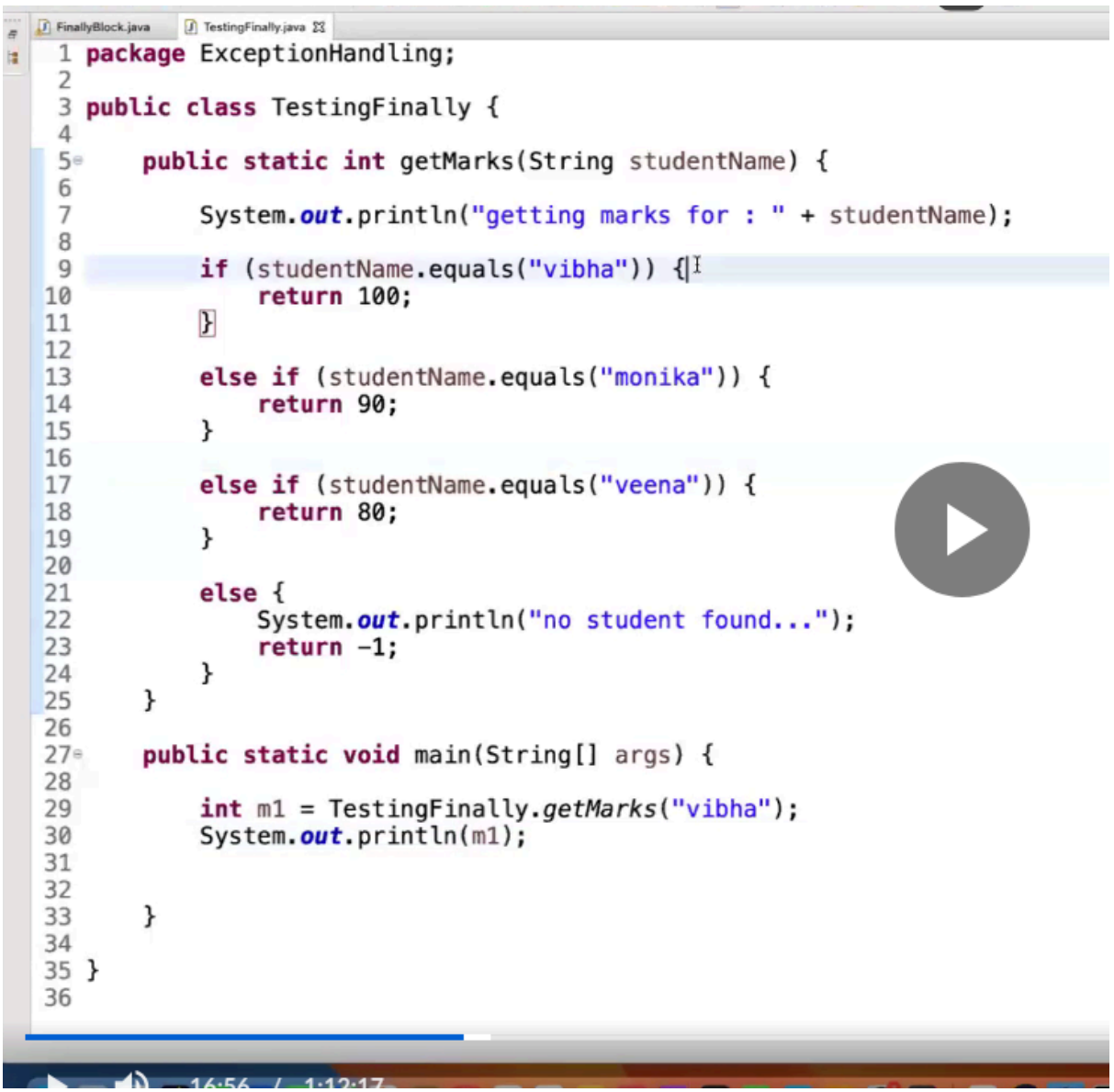
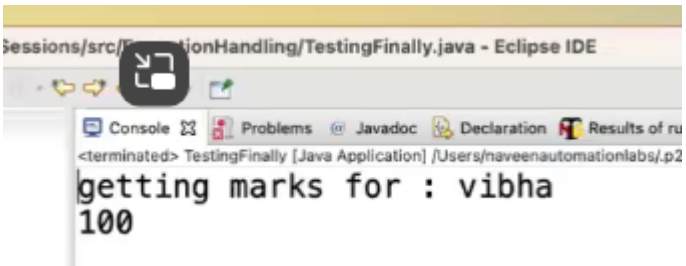
Anything can come in try catch finally block-



Finally block always runs irrespective of exception.

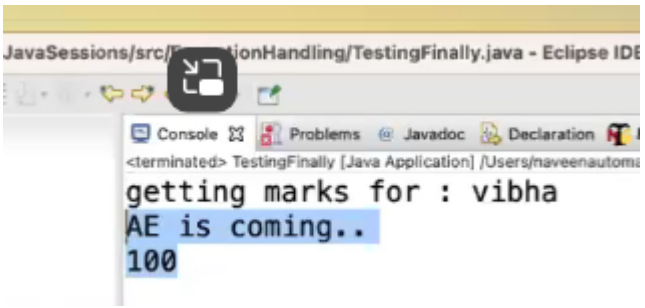
```
26
27          //use case:
28
29          //1. make the connection DB: un, pwd, ip:port: try-catch
30          //2. hit the SQL query: try-catch
31          //3. result from DB: try-catch
32          //4. display the data on UI: try-catch
33          //5. logout
34
35          //finally{
36                  //6. close connection with DB
37                  //}
38
39          //reading from excel/file
40
41
```





Add exception-

```
6
7           System.out.println("getting marks for : " + studentName);
8
9           if (studentName.equals("vibha")) {
10              try {
11                  int i = 9 / 0;
12              } catch (ArithmeticException e) {
13                  System.out.println("AE is coming..");
14              }
15              return 100;
16          }
```



paste testfinally1-

testfinally1.ja | testfinally3.ja | testfinally4.ja | testfinally5.ja | testfinally6.ja | testfinally7.ja | testfinally8.ja

```
1  package com.day24;
2
3  public class testfinally1 {
4
5      //method called getmarks.
6      //takes in studentname.
7      //method returns marks.
8      public static int getmarks(String studentname) {
9
10         System.out.println("getting marks for "+studentname);
11         if(studentname.equals("johny")) {
12             //add exception with return.
13             try {
14                 int i=9/0;
15             } catch (ArithmeticException e) {
16                 System.out.println("arithmetic exception .....");
17             }
18             return 100;
19         }else if(studentname.equals("rose")) {
20             return 90;
21         }else if(studentname.equals("tiger")) {
22             return 80;
23         }else {
24             System.out.println("no student found .........");
25             return -1;
26         }
27
28 //      System.out.println("end of the method");//Unreachable code
29     }
```

```
//      System.out.println("end of the method");//Unreachable code
    }

    public static void main(String[] args) {

        int marks=getmarks("johny");
        System.out.println(marks);

        int marks1=testfinally1.getmarks("rose");
        System.out.println(marks1);

        System.out.println("end of the program");

    }

}

//getting marks for johny
//arithmetic exception .............
//100
//getting marks for rose
//90
//end of the program
```

paste testfinally2-

testfinally2.java | testfinally3.java | testfinally4.java | testfinally5.java | testfinally6.java | testfinally7.java | testfir

```java
package com.day24;

public class testfinally2 {

    //method called getmarks.
    //takes in studentname.
    //method returns marks.
    public static int getmarks(String studentname) {

        System.out.println("getting marks for "+studentname);
        if(studentname.equals("johny")) {
            //return inside try lets see.
            try {
                int i=9/0;
                return 234234;
            } catch (ArithmeticException e) {
                System.out.println("arithmetic exception .....");
            }
            return 100;
        }else if(studentname.equals("rose")) {
            return 90;
        }else if(studentname.equals("tiger")) {
            return 80;
        }else {
            System.out.println("no student found .....");
            return -1;
        }

//      System.out.println("end of the method");//Unreachable code
    }

    public static void main(String[] args) {

        int marks=getmarks("johny");
        System.out.println(marks);

        int marks1=testfinally2.getmarks("rose");
        System.out.println(marks1);

        System.out.println("end of the program");

    }

}

//getting marks for johny
//arithmetic exception .....
//100
//getting marks for rose
//90
//end of the program
```

paste testfinally3-

testfinally3.java | testfinally4.java | testfinally5.java | testfinally6.java | testfinally7.java | testfinally8.java

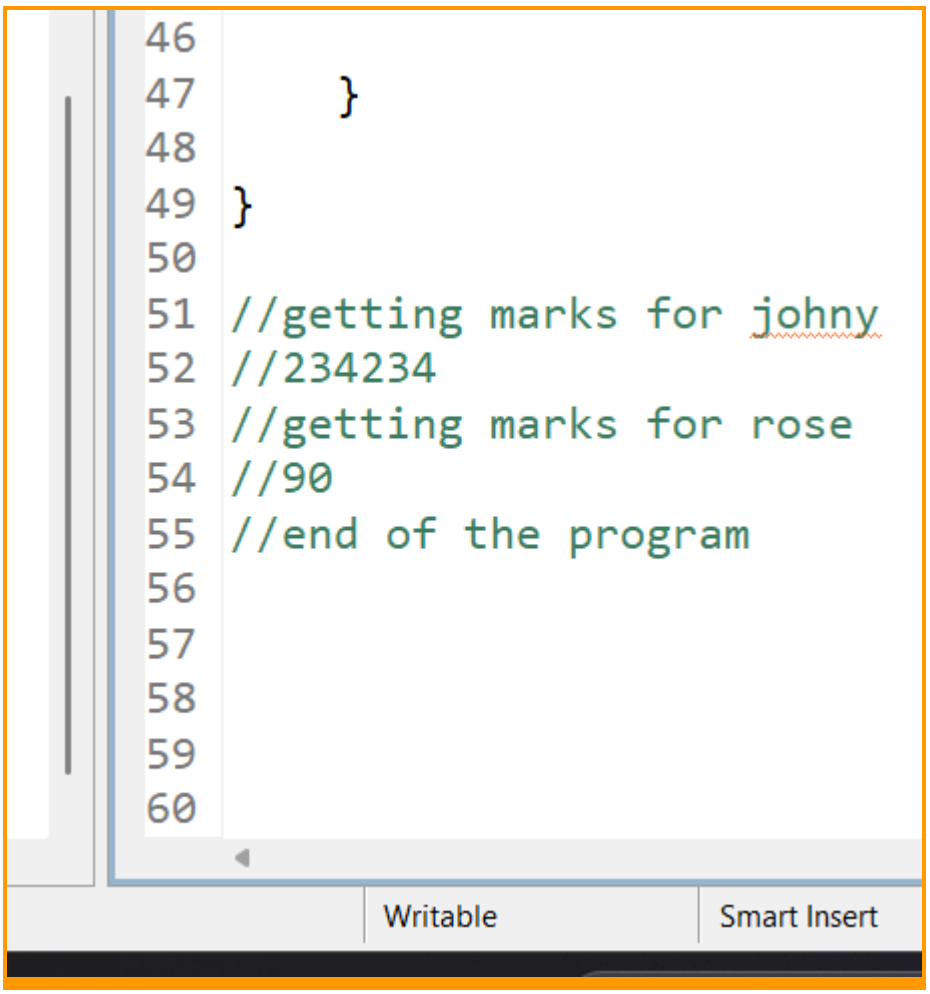
```java
package com.day24;

public class testfinally3 {

    //method called getmarks.
    //takes in studentname.
    //method returns marks.
    public static int getmarks(String studentname) {

        System.out.println("getting marks for "+studentname);
        if(studentname.equals("johny")) {
            //return inside try lets see.
            //no error.
            try {
                int i=9/3;
                return 234234;
            } catch (ArithmeticException e) {
                System.out.println("arithmetic exception .....");
            }
            return 100;
        }else if(studentname.equals("rose")) {
            return 90;
        }else if(studentname.equals("tiger")) {
            return 80;
        }else {
            System.out.println("no student found .....");
            return -1;
        }
```

```java
            return -1;
        }

//      System.out.println("end of the method");//Unreachable code
    }

    public static void main(String[] args) {

        int marks=getmarks("johny");
        System.out.println(marks);

        int marks1=testfinally3.getmarks("rose");
        System.out.println(marks1);

        System.out.println("end of the program");

    }

}

//getting marks for johny
//234234
//getting marks for rose
//90
//end of the program
```

paste testfinally4-

testfinally4.java | testfinally5.java | testfinally6.java | testfinally7.java | testfinally8.java | testfinally9.java

```java
package com.day24;

public class testfinally4 {

    //method called getmarks.
    //takes in studentname.
    //method returns marks.
    public static int getmarks(String studentname) {

        System.out.println("getting marks for "+studentname);
        if(studentname.equals("johny")) {
            //when we have return inside the catch block.
            //second return is unreachable.
            try {
                int i=9/0;
                return 234234;
            } catch (ArithmeticException e) {
                System.out.println("arithmetic exception .....");
                return 23434;
            }
            finally {
                System.out.println("finally block.....");
            }
            return 100;//Unreachable code
        }else if(studentname.equals("rose")) {
            return 90;
        }else if(studentname.equals("tiger")) {
            return 80;
        }else {
            return 80;
        }else {
            System.out.println("no student found .....");
            return -1;
        }

//      System.out.println("end of the method");//Unreachable code
    }

    public static void main(String[] args) {

        int marks=getmarks("johny");
        System.out.println(marks);

        int marks1=testfinally4.getmarks("rose");
        System.out.println(marks1);

        System.out.println("end of the program");

    }
```



paste testfinally5-

testfinally5.java | testfinally6.java | testfinally7.java | testfinally8.java | testfinally9.java

```java
package com.day24;

public class testfinally5 {

    //method called getmarks.
    //takes in studentname.
    //method returns marks.
    public static int getmarks(String studentname) {

        System.out.println("getting marks for "+studentname);
        if(studentname.equals("johny")) {
            //see the flow when catch has return.
            //first goes to the finally if present and then comes back to catch.
            //then runs all lines outside try catch finally.
            try {
                int i=9/0;
                return 234234;
            } catch (ArithmeticException e) {
                System.out.println("arithmetic exception ............");
                return 23434;
            }
            finally {
                System.out.println("finally block.............");
            }
//          return 100;//Unreachable code
        }else if(studentname.equals("rose")) {
            return 90;
        }else if(studentname.equals("tiger")) {
            return 80;
        }else if(studentname.equals("tiger")) {
            return 80;
        }else {
            System.out.println("no student found .........");
            return -1;
        }

//      System.out.println("end of the method");//Unreachable code
    }

    public static void main(String[] args) {

        int marks=getmarks("johny");
        System.out.println(marks);

        int marks1=testfinally5.getmarks("rose");
        System.out.println(marks1);

        System.out.println("end of the program");

    }
```

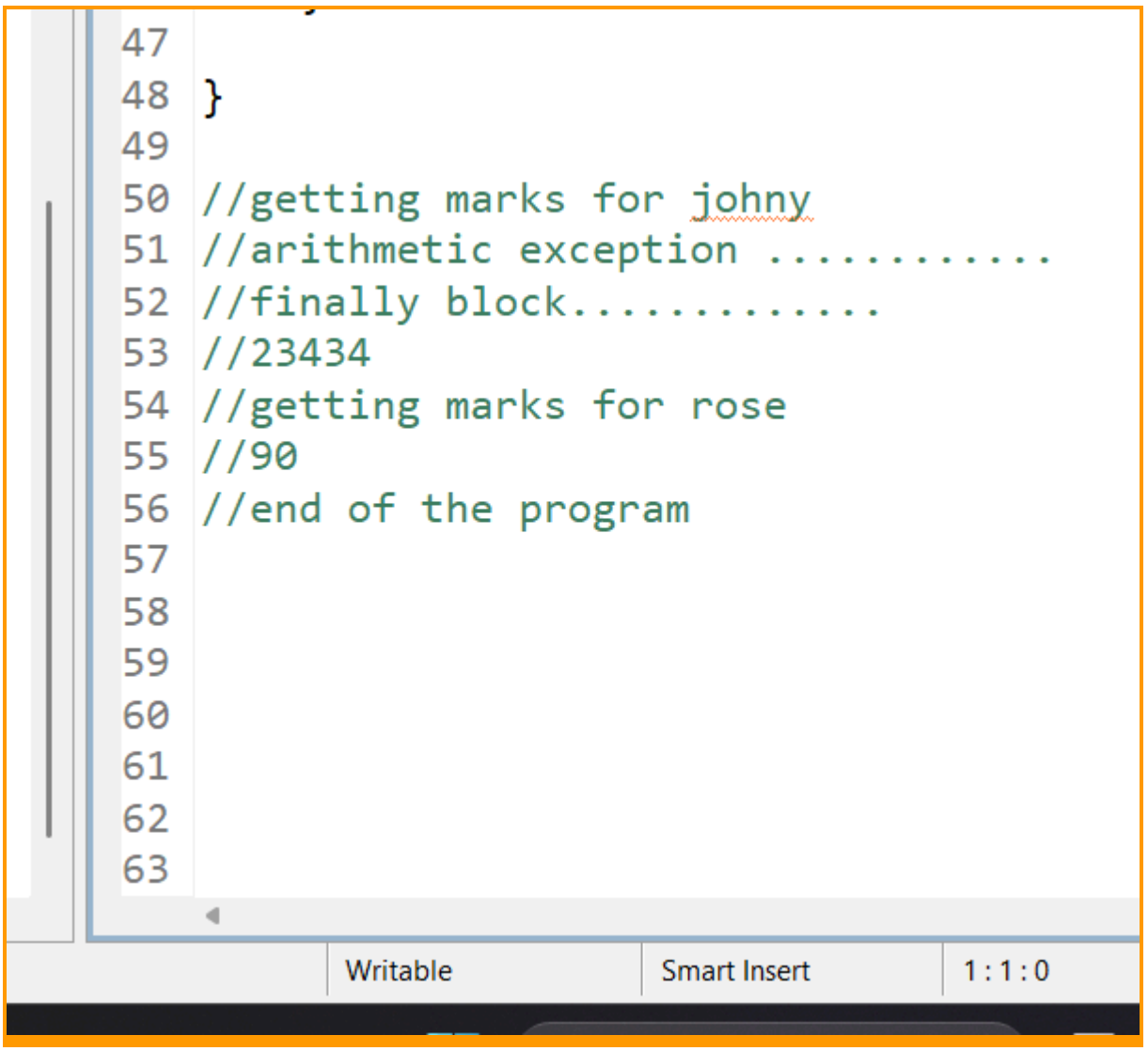
```
47
48      }
49
50  }
51
52  //getting marks for johny
53  //arithmetic exception .............
54  //finally block.............
55  //23434
56  //getting marks for rose
57  //90
58  //end of the program
59
```

paste testfinally6-

testfinally6.java | testfinally7.java | testfinally8.java | testfinally9.java

```
1  package com.day24;
2
3  public class testfinally6 {
4
5      //method called getmarks.
6      //takes in studentname.
7      //method returns marks.
8      public static int getmarks(String studentname) {
9
10         System.out.println("getting marks for "+studentname);
11         if(studentname.equals("johny")) {
12             //first finally runs then catch return runs.
13             //then rest of code runs.
14             try {
15                 int i=9/0;
16             } catch (ArithmeticException e) {
17                 System.out.println("arithmetic exception .............");
18                 return 23434;
19             }
20             finally {
21                 System.out.println("finally block.............");
22             }
23             return 100;
24         }else if(studentname.equals("rose")) {
25             return 90;
26         }else if(studentname.equals("tiger")) {
27             return 80;
28         }else {
```

```
27              return 80;
28          }else {
29              System.out.println("no student found .........");
30              return -1;
31          }
32
33  //      System.out.println("end of the method");//Unreachable code
34      }
35
36      public static void main(String[] args) {
37
38          int marks=getmarks("johny");
39          System.out.println(marks);
40
41          int marks1=testfinally6.getmarks("rose");
42          System.out.println(marks1);
43
44          System.out.println("end of the program");
45
46      }
47
48  }
```



paste testfinally7-

testfinally7.java | testfinally8.java | testfinally9.java

```java
package com.day24;

public class testfinally7 {

    //method called getmarks.
    //takes in studentname.
    //method returns marks.
    public static int getmarks(String studentname) {

        System.out.println("getting marks for "+studentname);
        if(studentname.equals("johny")) {
            //finally with return.
            try {
                int i=9/0;
            } catch (ArithmeticException e) {
                System.out.println("arithmetic exception .....");
                return 23434;
            }
            //warning when we add return to finally.
            //finally block does not complete normally
            //catch return will run.
            //but finally return will update the return value from catch.
            finally {
                System.out.println("finally block.....");
                return 67890;
            }
//          return 100;//Unreachable code
        }else if(studentname.equals("rose")) {
            return 90;
        }else if(studentname.equals("tiger")) {
            return 80;
        }else {
            System.out.println("no student found .....");
            return -1;
        }

//      System.out.println("end of the method");//Unreachable code
    }

    public static void main(String[] args) {

        int marks=getmarks("johny");
        System.out.println(marks);

        int marks1=testfinally7.getmarks("rose");
        System.out.println(marks1);

        System.out.println("end of the program");
```

```
47
48          System.out.println("end of the program");
49
50      }
51
52  }
53
54  //getting marks for johny
55  //arithmetic exception .............
56  //finally block..............
57  //67890
58  //getting marks for rose
59  //90
60  //end of the program
61
```

paste testfinally8-

testfinally8.java testfinally9.java

```
1  package com.day24;
2
3  public class testfinally8 {
4
5      //method called getmarks.
6      //takes in studentname.
7      //method returns marks.
8      public static int getmarks(String studentname) {
9
10         System.out.println("getting marks for "+studentname);
11         if(studentname.equals("johny")) {
12             //no exception anywhere.
13             try {
14                 int i=9/3;
15                 return 10;
16             } catch (ArithmeticException e) {
17                 System.out.println("arithmetic exception .............");
18                 return 23434;
19             }
20             finally {
21                 System.out.println("finally block..............");
22                 return 67890;
23             }
24 //          return 100;//Unreachable code
25         }else if(studentname.equals("rose")) {
26             return 90;
27         }else if(studentname.equals("tiger")) {
28             return 80;
29         }else {
```

```
27          }else if(studentname.equals("tiger")) {
28              return 80;
29          }else {
30              System.out.println("no student found .........");
31              return -1;
32          }
33
34  //      System.out.println("end of the method");//Unreachable code
35      }
36
37      public static void main(String[] args) {
38
39          int marks=getmarks("johny");
40          System.out.println(marks);
41
42          int marks1=testfinally8.getmarks("rose");
43          System.out.println(marks1);
44
45          System.out.println("end of the program");
46
47      }
```

```
46
47          }
48
49  }
50
51  //getting marks for johny
52  //finally block.............
53  //67890
54  //getting marks for rose
55  //90
56  //end of the program
57
```

paste testfinally9-

testfinally9.java

```java
package com.day24;

public class testfinally9 {

    //method called getmarks.
    //takes in studentname.
    //method returns marks.
    public static int getmarks(String studentname) {

        System.out.println("getting marks for "+studentname);
        if(studentname.equals("johny")) {
            //no return in finally.
            //no exception.
            //return statement runs after finally block only.
            try {
                int i=9/3;
                return 10;
            } catch (ArithmeticException e) {
                System.out.println("arithmetic exception ............");
                return 23434;
            }
            finally {
                System.out.println("finally block.............");
            }
//          return 100;//Unreachable code
        }else if(studentname.equals("rose")) {
            return 90;
        }else if(studentname.equals("tiger")) {
            return 80;
        }else {
            System.out.println("no student found .........");
            return -1;
        }

//      System.out.println("end of the method");//Unreachable code
    }

    public static void main(String[] args) {

        int marks=getmarks("johny");
        System.out.println(marks);

        int marks1=testfinally9.getmarks("rose");
        System.out.println(marks1);

        System.out.println("end of the program");
```

```
45
46          System.out.println("end of the program");
47
48      }
49
50  }
51
52  //getting marks for johny
53  //finally block.............
54  //10
55  //getting marks for rose
56  //90
57  //end of the program
58
```

Return statement in catch-

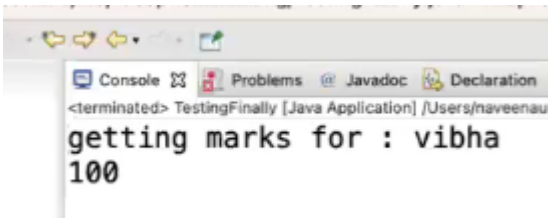
```
7          System.out.println("getting marks for : " + studentName);
8
9          if (studentName.equals("vibha")) {
10             try {
11                 int i = 9 / 0;
12             } catch (ArithmeticException e) {
13                 System.out.println("AE is coming..");
14                 return 10;
15             }
16             return 100;
17         }
```

```
...ssions/src/...ionHandling/TestingFinally.java - Eclipse
Console  Problems  Javadoc  Declaration
<terminated> TestingFinally [Java Application] /Users/naveenaut
getting marks for : vibha
AE is coming..
10
```

Return is last statement in the block to run.

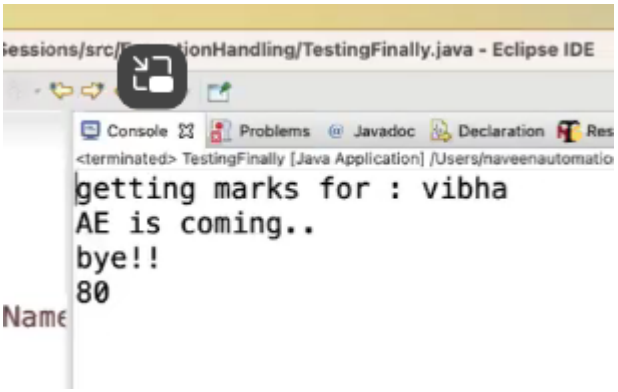
No exception-

```
6
7       System.out.println("getting marks for : " + studentName);
8
9       if (studentName.equals("vibha")) {
10          try {
11              int i = 9 / 3;
12          } catch (ArithmeticException e) {
13              System.out.println("AE is coming..");
14              return 10;
15          }
16          return 100;
17      }
```



Finally with return-

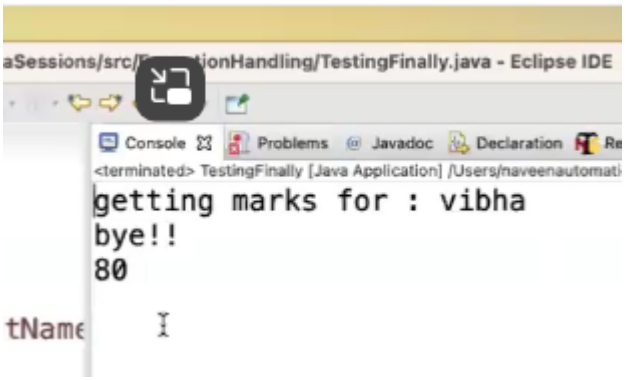
```
5   public static int getMarks(String studentName) {
6
7       System.out.println("getting marks for : " + studentName);
8
9       if (studentName.equals("vibha")) {
10          try {
11              int i = 9 / 0;
12          } catch (ArithmeticException e) {
13              System.out.println("AE is coming..");
14              return 10;
15          }
16          finally {
17              System.out.println("bye!!");
18              return 80;
19          }
20      }
```



Finally runs even after return statement. see above example, though catch has return but still finally runs.

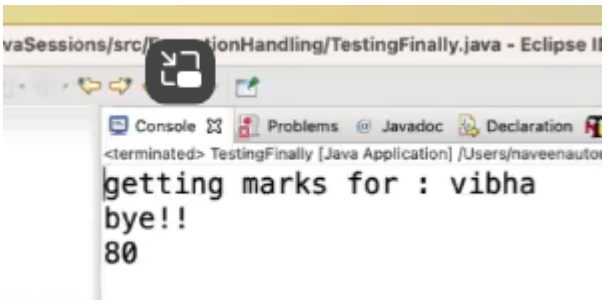
Without exception-

```
5    public static int getMarks(String studentName) {
6
7        System.out.println("getting marks for : " + studentName);
8
9        if (studentName.equals("vibha")) {
10           try {
11               int i = 9 / 3;
12           }
13           catch (ArithmeticException e) {
14               System.out.println("AE is coming..");
15               return 10;
16           }
17           finally {
18               System.out.println("bye!!");
19               return 80;
20           }
21       }
```



Some more code changes-

```
5    public static int getMarks(String studentName) {
6
7        System.out.println("getting marks for : " + studentName);
8
9        if (studentName.equals("vibha")) {
10           try {
11               int i = 9 / 3;
12               return 30;
13           }
14           catch (ArithmeticException e) {
15               System.out.println("AE is coming..");
16               return 10;
17           }
18           finally {
19               System.out.println("bye!!");
20               return 80;
21           }
22       }
```



Some more code changes-

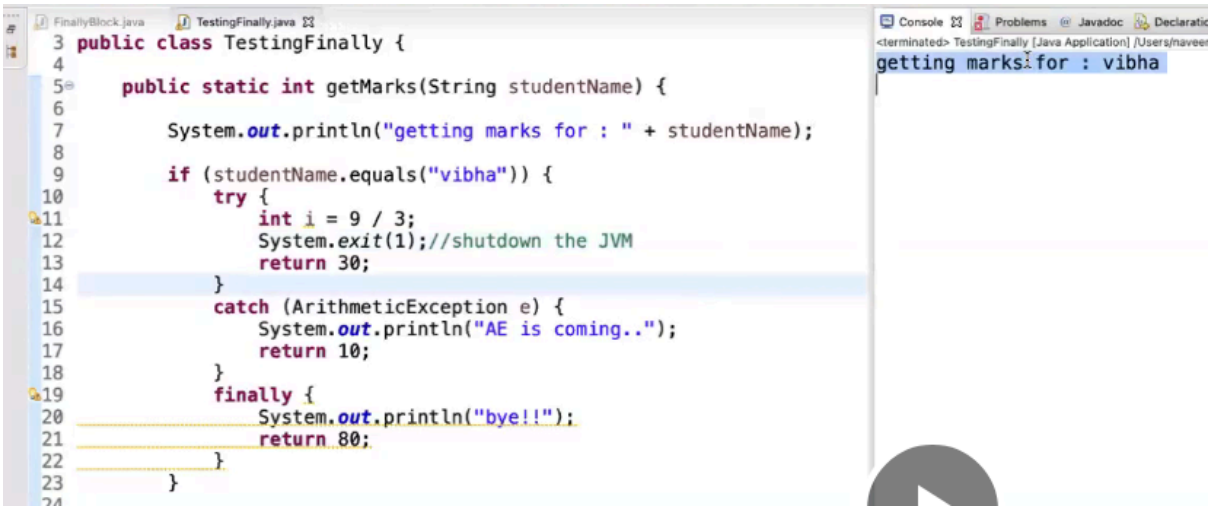
```
4
5    public static int getMarks(String studentName) {
6
7        System.out.println("getting marks for : " + studentName);
8
9        if (studentName.equals("vibha")) {
10           try {
11               int i = 9 / 3;
12               return 30;
13           }
14           catch (ArithmeticException e) {
15               System.out.println("AE is coming..");
16               return 10;
17           }
18           finally {
19               System.out.println("bye!!");
20               //return 80;
21           }
22       }
```

```
Sessions/src/ExceptionHandling/TestingFinally.java - Eclip
Console  Problems  Javadoc  Declaratio
<terminated> TestingFinally [Java Application] /Users/naveen
getting marks for : vibha
bye!!
30
```

Unreachable return-

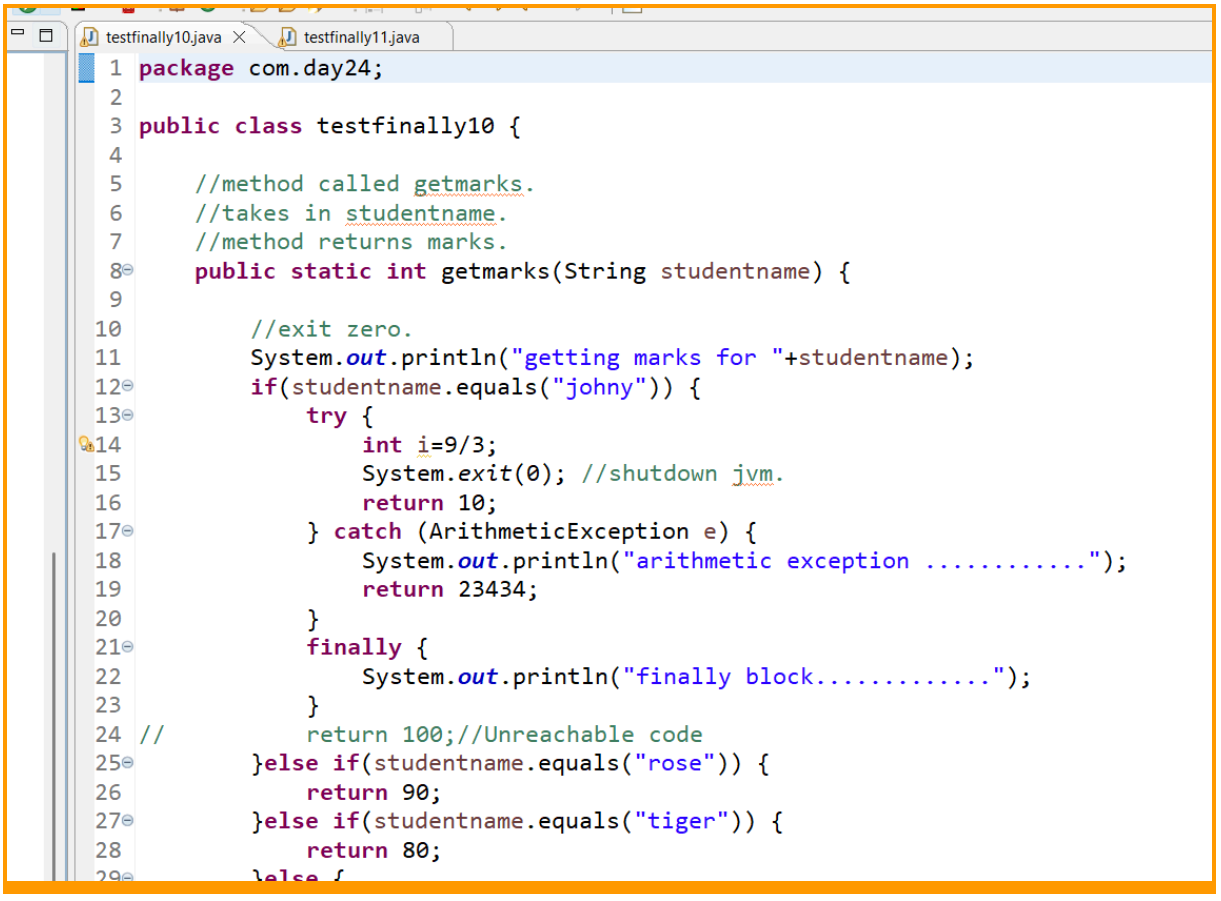
```
5    public static int getMarks(String studentName) {
6
7        System.out.println("getting marks for : " + studentName);
8
9        if (studentName.equals("vibha")) {
10           try {
11               int i = 9 / 3;
12               return 30;
13           }
14           catch (ArithmeticException e) {
15               System.out.println("AE is coming..");
16               return 10;
17           }
18           finally {
19               System.out.println("bye!!");
20               //return 80;
21           }
22           return 50;
23       }
```

System exit –



It will terminate and no more statements run.

paste testfinally10-



```
        }else if(studentname.equals("tiger")) {
            return 80;
        }else {
            System.out.println("no student found .........");
            return -1;
        }

//      System.out.println("end of the method");//Unreachable code
    }

    public static void main(String[] args) {

        int marks=getmarks("johny");
        System.out.println(marks);

        int marks1=testfinally10.getmarks("rose");
        System.out.println(marks1);

        System.out.println("end of the program");
```

```
            System.out.println("end of the program");

        }

}

//getting marks for johny
```

paste test finally11-

testfinally11.java

```java
package com.day24;

public class testfinally11 {

    //method called getmarks.
    //takes in studentname.
    //method returns marks.
    public static int getmarks(String studentname) {

        //exit one.
        //zero and one works same way.
        System.out.println("getting marks for "+studentname);
        if(studentname.equals("johny")) {
            try {
                int i=9/3;
                System.exit(1); //shutdown jvm.
                return 10;
            } catch (ArithmeticException e) {
                System.out.println("arithmetic exception .....");
                return 23434;
            }
            finally {
                System.out.println("finally block.....");
            }
//          return 100;//Unreachable code
        }else if(studentname.equals("rose")) {
            return 90;
        }else if(studentname.equals("tiger")) {
            return 80;
        }else if(studentname.equals("tiger")) {
            return 80;
        }else {
            System.out.println("no student found .....");
            return -1;
        }

//      System.out.println("end of the method");//Unreachable code
    }

    public static void main(String[] args) {

        int marks=getmarks("johny");
        System.out.println(marks);

        int marks1=testfinally11.getmarks("rose");
        System.out.println(marks1);

        System.out.println("end of the program");
```

```
45
46          System.out.println("end of the program");
47
48      }
49
50  }
51
52  //getting marks for johny
53
```

When will exception block come with exit-

```
3  public class TestingFinally {
4
5      public static int getMarks(String studentName) {
6
7          System.out.println("getting marks for : " + studentName);
8
9          if (studentName.equals("vibha")) {
10             try {
11                 int i = 9 / 0;
12                 System.exit(1);//shutdown the JVM
13                 return 30;
14             }
15             catch (ArithmeticException e) {
16                 System.out.println("AE is coming..");
17                 return 10;
18             }
19             finally {
20                 System.out.println("bye!!");
21                 return 80;
22             }
23         }
24
```

```
Console  Problems  Javadoc  Declaration
<terminated> TestingFinally [Java Application] /Users/naveenautoma
getting marks for : vibha
AE is coming..
bye!!
80
```

paste testfinally12-

testfinally12.java | testfinally13.java | testfinally14.java | testfinally15.java

```java
package com.day24;

public class testfinally12 {

    //method called getmarks.
    //takes in studentname.
    //method returns marks.
    public static int getmarks(String studentname) {

        //exit zero.
        //when exception, exit statement is not reached and programs as usual.
        //here return of catch is taken after finally runs.
        System.out.println("getting marks for "+studentname);
        if(studentname.equals("johny")) {
            try {
                int i=9/0;
                System.exit(0); //shutdown jvm.
                return 10;
            } catch (ArithmeticException e) {
                System.out.println("arithmetic exception ............");
                return 23434;
            }
            finally {
                System.out.println("finally block.............");
            }
//          return 100;//Unreachable code
        }else if(studentname.equals("rose")) {
            return 90;
        }else if(studentname.equals("tiger")) {
            return 80;
        }else {
            System.out.println("no student found .........");
            return -1;
        }

//      System.out.println("end of the method");//Unreachable code
    }

    public static void main(String[] args) {

        int marks=getmarks("johny");
        System.out.println(marks);

        int marks1=testfinally12.getmarks("rose");
        System.out.println(marks1);

        System.out.println("end of the program");
```

```
46
47          System.out.println("end of the program");
48
49      }
50
51  }
52
53  //getting marks for johny
54  //arithmetic exception .............
55  //finally block..............
56  //23434
57  //getting marks for rose
58  //90
59  //end of the program
60
```

paste testfinally13-

testfinally13.java | testfinally14.java | testfinally15.java

```
1  package com.day24;
2
3  public class testfinally13 {
4
5      //method called getmarks.
6      //takes in studentname.
7      //method returns marks.
8      public static int getmarks(String studentname) {
9
10         //exit zero.
11         //when exception, exit statement is not reached and programs as usual.
12         //here return of finally taken.
13         System.out.println("getting marks for "+studentname);
14         if(studentname.equals("johny")) {
15             try {
16                 int i=9/0;
17                 System.exit(0); //shutdown jvm.
18                 return 10;
19             } catch (ArithmeticException e) {
20                 System.out.println("arithmetic exception .............");
21                 return 23434;
22             }
23             finally {
24                 System.out.println("finally block..............");
25                 return 10;
26             }
27 //          return 100;//Unreachable code
28         }else if(studentname.equals("rose")) {
29             return 90;
```

```
28          }else if(studentname.equals("rose")) {
29              return 90;
30          }else if(studentname.equals("tiger")) {
31              return 80;
32          }else {
33              System.out.println("no student found .........");
34              return -1;
35          }
36
37  //      System.out.println("end of the method");//Unreachable code
38      }
39
40      public static void main(String[] args) {
41
42          int marks=getmarks("johny");
43          System.out.println(marks);
44
45          int marks1=testfinally13.getmarks("rose");
46          System.out.println(marks1);
47
48          System.out.println("end of the program");
49
50      }
51
```

```
49
50      }
51
52  }
53
54  //getting marks for johny
55  //arithmetic exception .............
56  //finally block.............
57  //10
58  //getting marks for rose
59  //90
60  //end of the program
61
```

paste testfinally14-

testfinally14.java | testfinally15.java

```java
package com.day24;

public class testfinally14 {

    //method called getmarks.
    //takes in studentname.
    //method returns marks.
    public static int getmarks(String studentname) {

        //exit one.
        //when exception, exit statement is not reached and programs as usual.
        //here return of catch is taken after finally runs.
        System.out.println("getting marks for "+studentname);
        if(studentname.equals("johny")) {
            try {
                int i=9/0;
                System.exit(0); //shutdown jvm.
                return 10;
            } catch (ArithmeticException e) {
                System.out.println("arithmetic exception ............");
                return 23434;
            }
            finally {
                System.out.println("finally block.............");
            }
//          return 100;//Unreachable code
        }else if(studentname.equals("rose")) {
            return 90;
        }else if(studentname.equals("tiger")) {
            return 80;
        }else {
            System.out.println("no student found .........");
            return -1;
        }

//      System.out.println("end of the method");//Unreachable code
    }

    public static void main(String[] args) {

        int marks=getmarks("johny");
        System.out.println(marks);

        int marks1=testfinally14.getmarks("rose");
        System.out.println(marks1);

        System.out.println("end of the program");
```

```
46
47          System.out.println("end of the program");
48
49      }
50
51  }
52
53  //getting marks for johny
54  //arithmetic exception .............
55  //finally block..............
56  //23434
57  //getting marks for rose
58  //90
59  //end of the program
60
```

paste testfinally15-

testfinally15.java

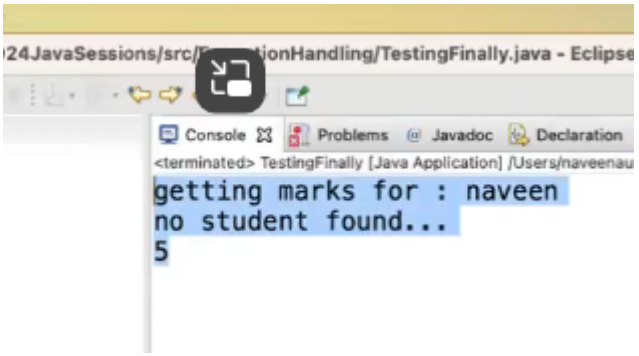
```
1  package com.day24;
2
3  public class testfinally15 {
4
5      //method called getmarks.
6      //takes in studentname.
7      //method returns marks.
8      public static int getmarks(String studentname) {
9
10         //exit one.
11         //when exception, exit statement is not reached and programs as usual.
12         //here return of finally taken.
13         System.out.println("getting marks for "+studentname);
14         if(studentname.equals("johny")) {
15             try {
16                 int i=9/0;
17                 System.exit(0); //shutdown jvm.
18                 return 10;
19             } catch (ArithmeticException e) {
20                 System.out.println("arithmetic exception .............");
21                 return 23434;
22             }
23             finally {
24                 System.out.println("finally block..............");
25                 return 10;
26             }
```

```
25                  return 10;
26              }
27  //          return 100;//Unreachable code
28          }else if(studentname.equals("rose")) {
29              return 90;
30          }else if(studentname.equals("tiger")) {
31              return 80;
32          }else {
33              System.out.println("no student found .........");
34              return -1;
35          }
36
37  //      System.out.println("end of the method");//Unreachable code
38      }
39
40      public static void main(String[] args) {
41
42          int marks=getmarks("johny");
43          System.out.println(marks);
44
45          int marks1=testfinally15.getmarks("rose");
46          System.out.println(marks1);
47
48          System.out.println("end of the program");
49
50      }
```

```
49
50          }
51
52  }
53
54  //getting marks for johny
55  //arithmetic exception .............
56  //finally block..............
57  //10
58  //getting marks for rose
59  //90
60  //end of the program
61
```

Try finally without catch-

```
21          }
22
23          else if (studentName.equals("monika")) {
24              return 90;
25          }
26
27          else if (studentName.equals("veena")) {
28              return 40;
29          }
30
31          else {
32              System.out.println("no student found...");
33
34              try {
35                  int p = 9 / 0;
36              } finally {
37                  return 5;
38              }
39
40          }
41      }
42
43      public static void main(String[] args) {
44
45          int m1 = TestingFinally.getMarks("naveen");
46          System.out.println(m1);
47
48      }
49
```



paste testfinally16-

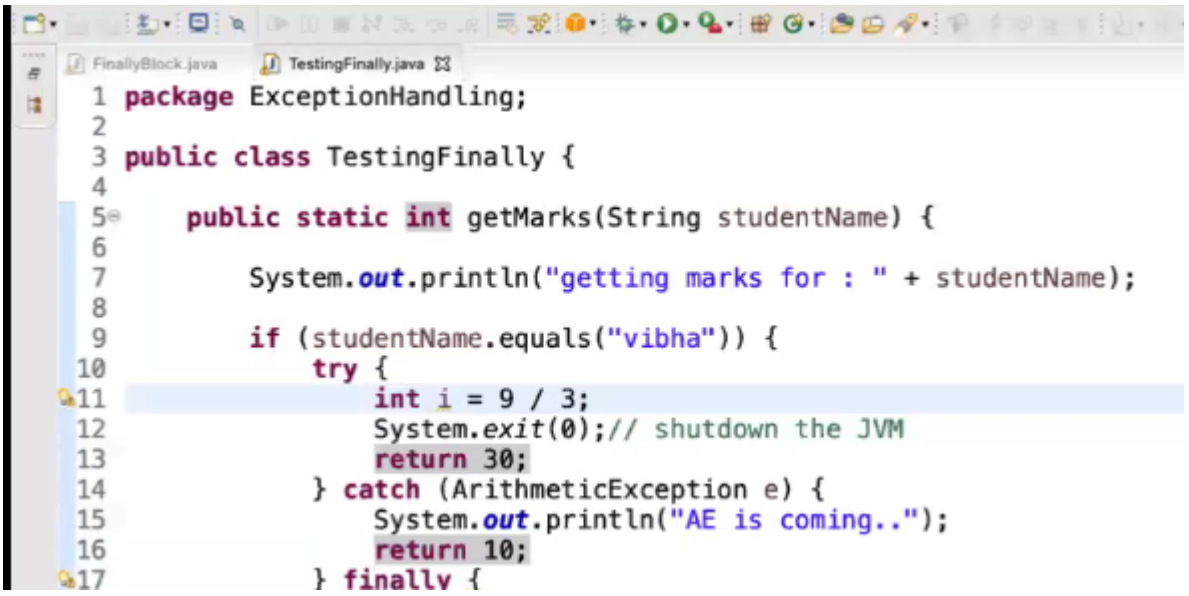
testfinally16.java

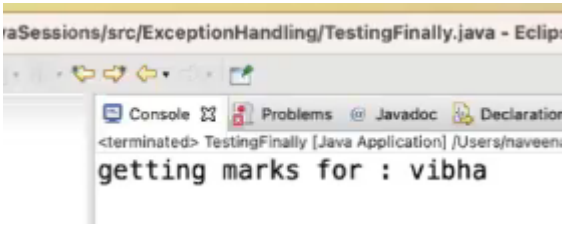
```java
package com.day24;

public class testfinally16 {

    //method called getmarks.
    //takes in studentname.
    //method returns marks.
    public static int getmarks(String studentname) {
        //try finally without catch.

        System.out.println("getting marks for "+studentname);
        if(studentname.equals("johny")) {
            return 100;
        }else if(studentname.equals("rose")) {
            return 90;
        }else if(studentname.equals("tiger")) {
            return 80;
        }else {
            System.out.println("no student found .........");
            try {
                int p=9/0;
            }finally {//finally block does not complete normally
                //warning message.
                return 5;
            }
        }
            }
        }

//      System.out.println("end of the method");//Unreachable code
    }

    public static void main(String[] args) {

        int marks=getmarks("karan");
        System.out.println(marks);

        int marks1=testfinally16.getmarks("rose");
        System.out.println(marks1);

        System.out.println("end of the program");

    }

}
```

```
42
43 }
44
45 //getting marks for karan
46 //no student found .........
47 //5
48 //getting marks for rose
49 //90
50 //end of the program
51
```

Exit 0-

Partial shutdown.





Exit 1 is full shutdown.

Exit 0 is partial shutdown.

AccessModifiersinJava (3).png

	default	private	protected	public
Same Class	Yes	Yes	Yes	Yes
Same package subclass	Yes	No	Yes	Yes
Same package non-subclass	Yes	No	Yes	Yes
Different package subclass	No	No	Yes	Yes
Different package non-subclass	No	No	No	Yes

Car.java

```java
package AccMod1;

public class Car {

    String name;
    public int price;
    protected String color;
    private String chasisNum;


    public static void main(String[] args) {


        Car c = new Car();
        c.chasisNum = "123";


    }

}
```

paste car1-

car1.java bmw1.java

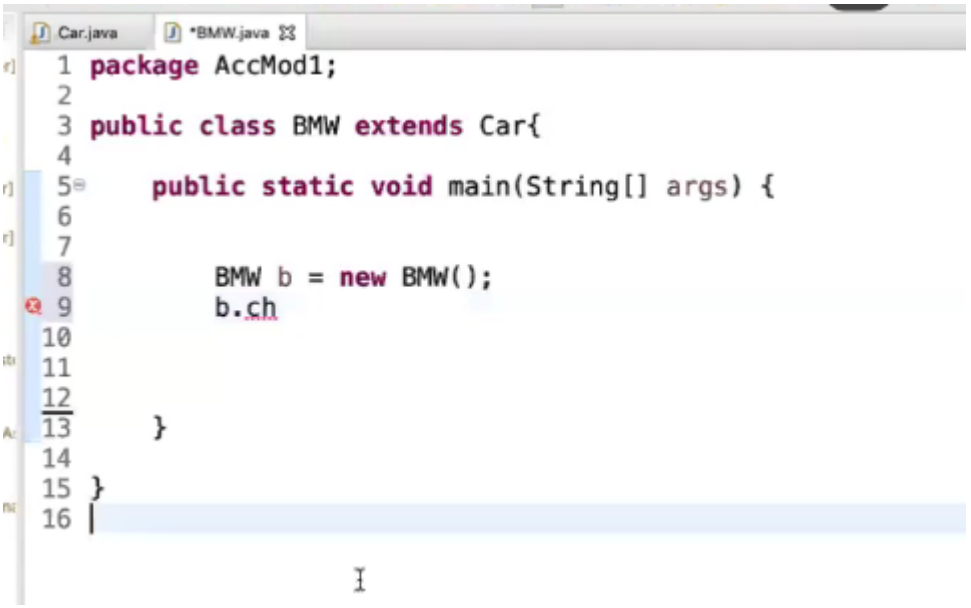
```java
package com.day24;

public class car1 {

    String name;
    public int price;
    protected String color;
    private String chasisnumber;

    public static void main(String[] args) {

        car1 c=new car1();
        String s1=c.chasisnumber;
        System.out.println(s1);
    }

    private int getname(String name) {
        System.out.println("private method get name");
        return 10;
    }

}

//null
```

paste bmw1-

car1.java bmw1.java

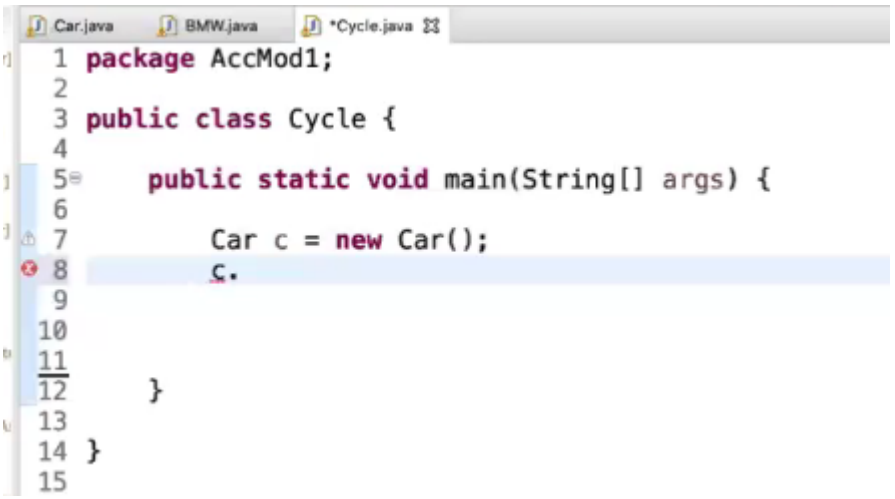
```java
package com.day24;

public class bmw1  extends car1{

    public static void main(String[] args) {

        bmw1 b=new bmw1();
//      b.//cannot get private variables or methods outside class.
//      b.
    }
}
```

Second row example-



Third row example-

cannot access private variables and methods outside class even object of car created.



Fourth row example-

Car.java BMW.java Cycle.java *Audi.java

```
1  package AccMod2;
2
3  import AccMod1.Car;
4
5  public class Audi extends Car{
6
7      public static void main(String[] args) {
8
9
10
11         Audi a = new Audi();
12
13         a.
14
15     }
16
17 }
18
```

Fifth row example-

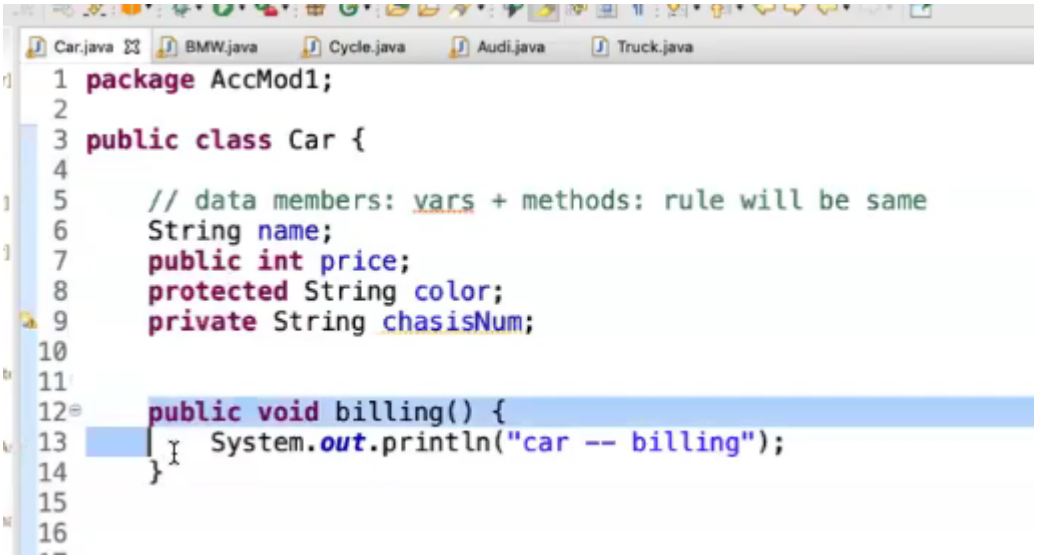
Car.java BMW.java Cycle.java Audi.java Truck.java

```
1  package AccMod2;
2
3  import AccMod1.Car;
4
5  public class Truck {
6
7      public static void main(String[] args) {
8
9
10         Car c = new Car();
11
12         c.price = 100;
13
14     }
15
16 }
17
```

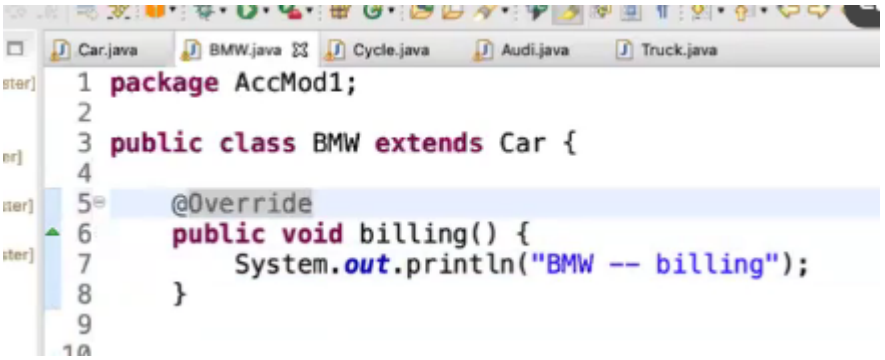
Car.java BMW.java Cycle.java Audi.java Truck.java

```
1  package AccMod1;
2
3  public class Car {
4
5
6      //data members: vars + methods: rule will be same
7      String name;
8      public int price;
9      protected String color;
10     private String chasisNum;
11
```

Public methods can be overridden-

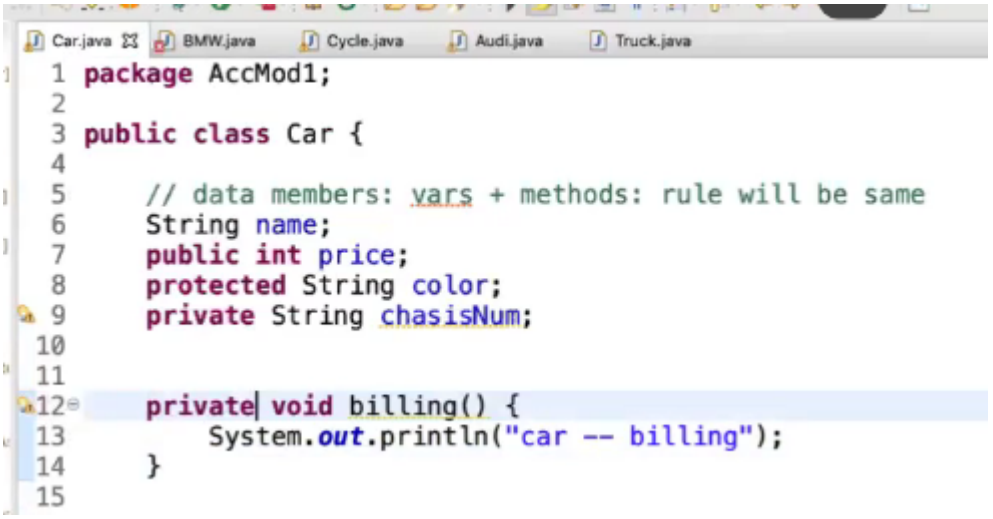


Bmw-



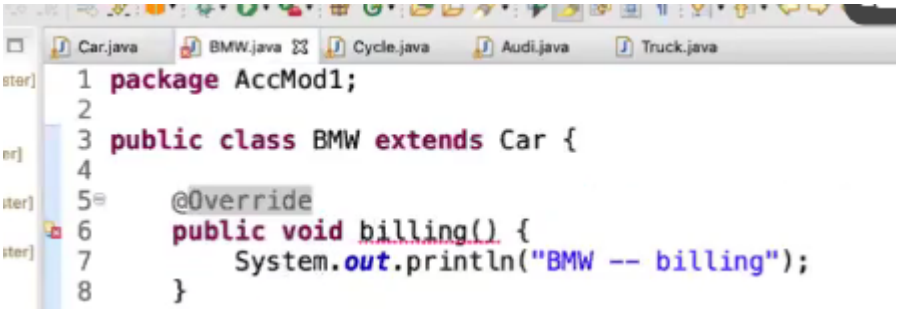
Private methods cannot be overridden-

we cant access only outside class.

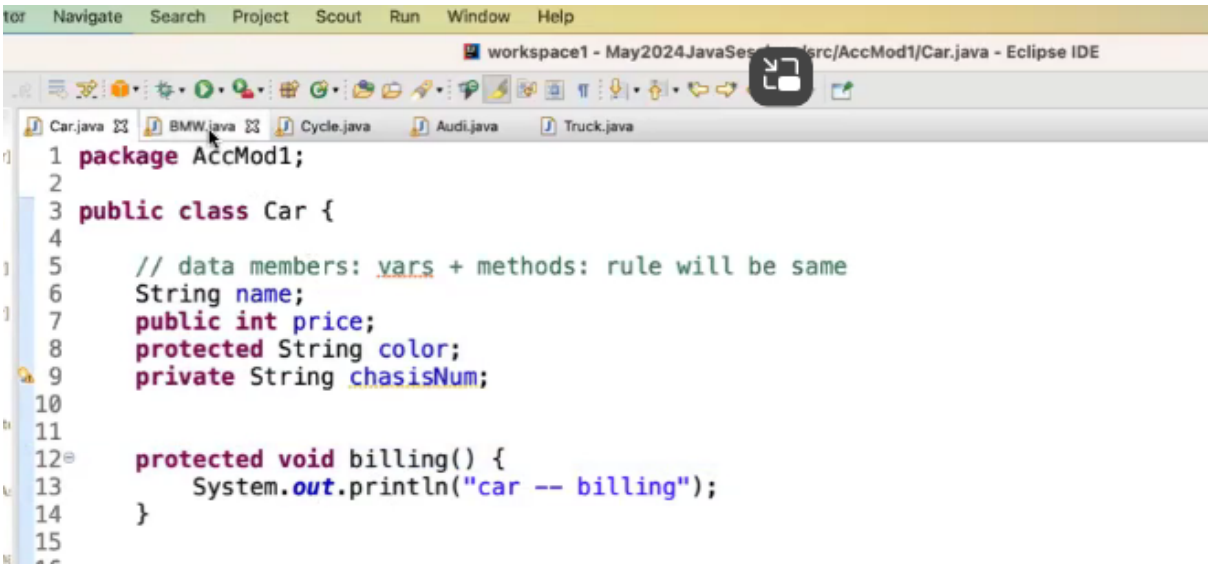


Bmw error-

//The method getname(String) of type bmw3
//must override or implement a supertype method



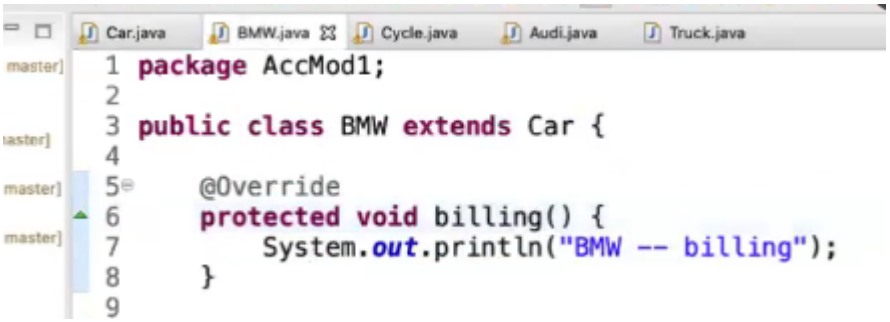
Protected-



Always go towards right.

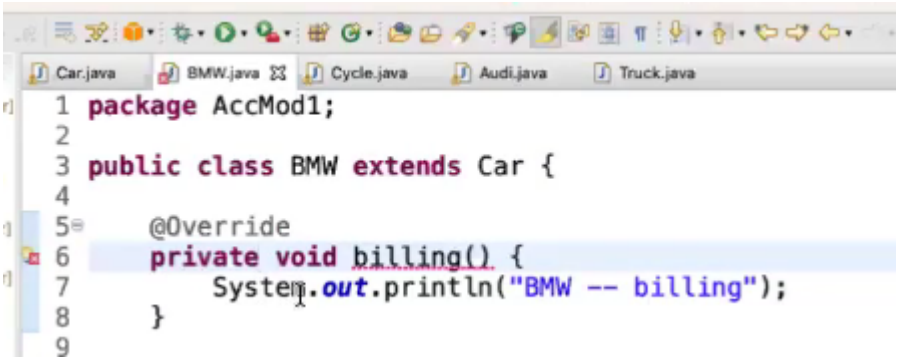
If parent class has protected, then in child class can have public or protected only.

Bmw-



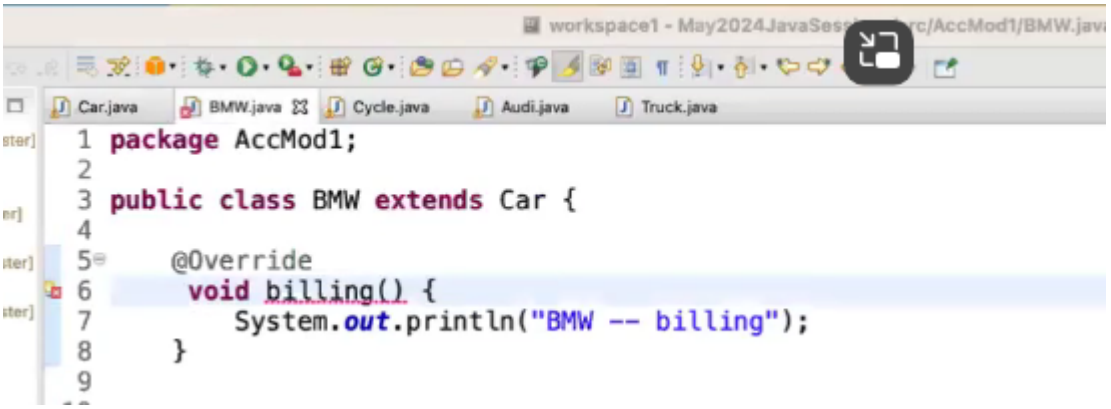
Private not allowed-

// Cannot reduce the visibility of the inherited method from car2



Default not allowed in bmw-

// Cannot reduce the visibility of the inherited method from car2



protected can become public-

paste car2-

bmw7.java car2.java

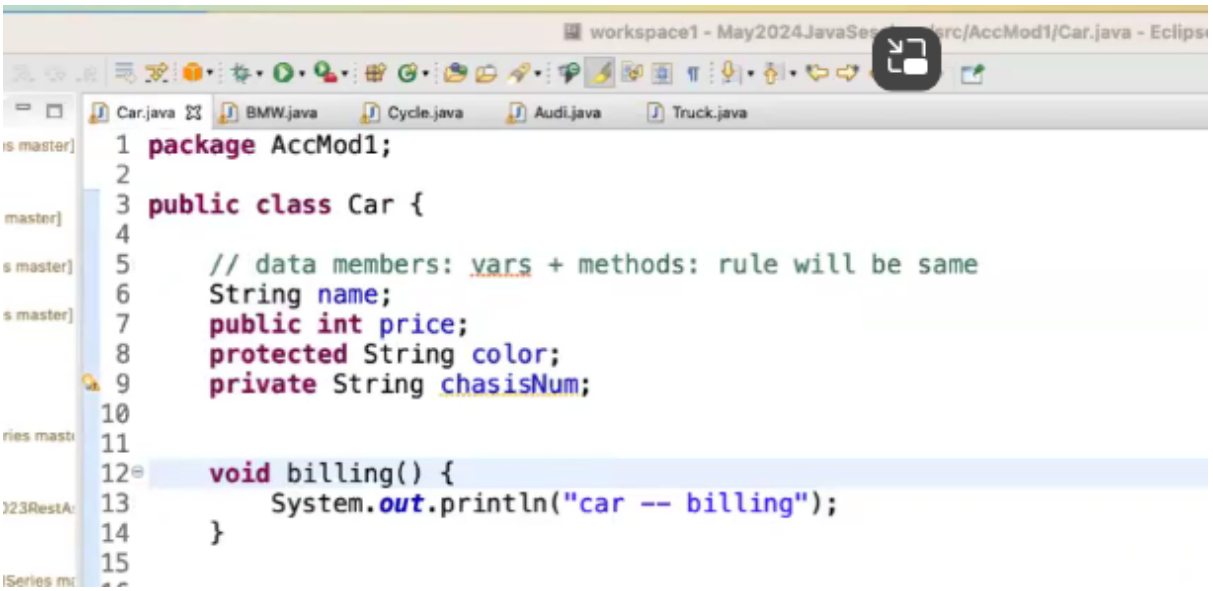
```java
package com.day24;

public class car2 {

    String name;
    public int price;
    protected String color;
    private String chasisnumber;



    private int getname(String name) {
        System.out.println("private method get name");
        return 10;
    }

    public String billing(int a) {
        System.out.println("car billing method");
        return "billing method";
    }

    protected void carname(String a) {
        System.out.println("protected method inside car");
    }

    int getlocation(String name) {
        System.out.println("get location method "+name);
        return 10;
    }
```

```java
        return 10;
    }

}
```

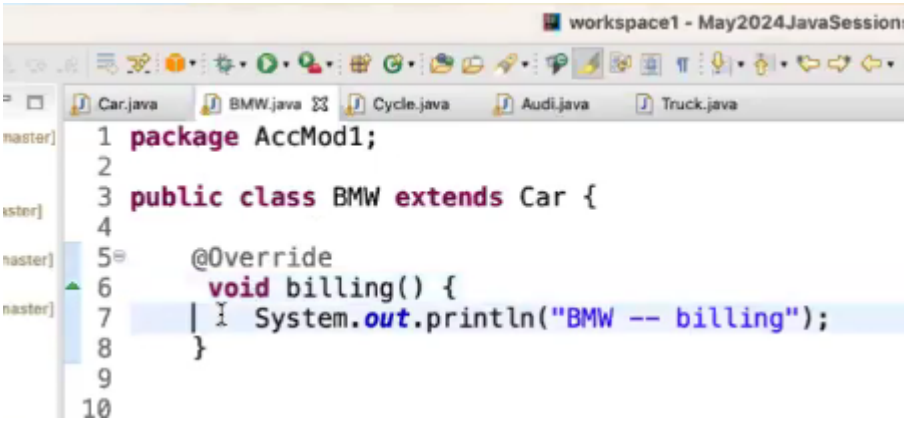
paste bmw7-

```java
package com.day24;

public class bmw7  extends car2{

    public static void main(String[] args) {

//      Always go towards right.
//      If parent class has protected, then in child class can have public or protected only.


        bmw7 b1=new bmw7();
        String s1=b1.color;
        System.out.println(s1);
        b1.carname("karan");
    }
    //protected can become public in car.
    @Override
    public void carname(String a) {
        System.out.println("protected method inside child class bmw");
    }

}

//null
//protected method inside child class bmw
```

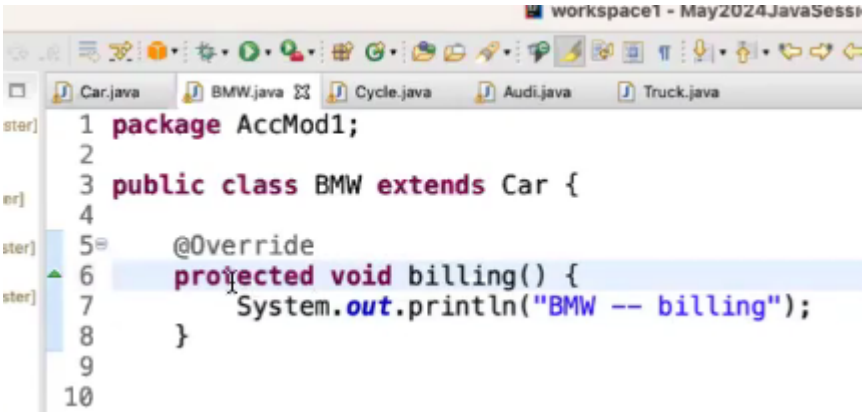
Default in parent class-



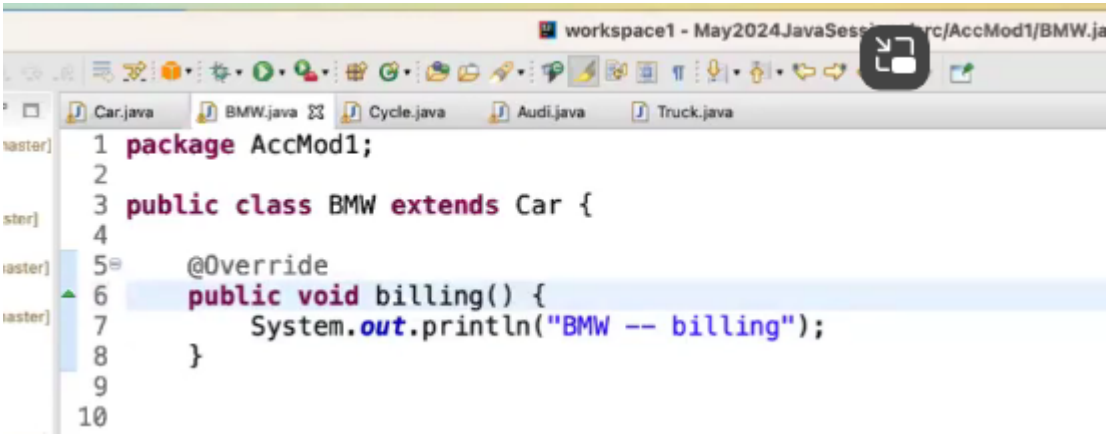
Child can have default-



Protected allowed-

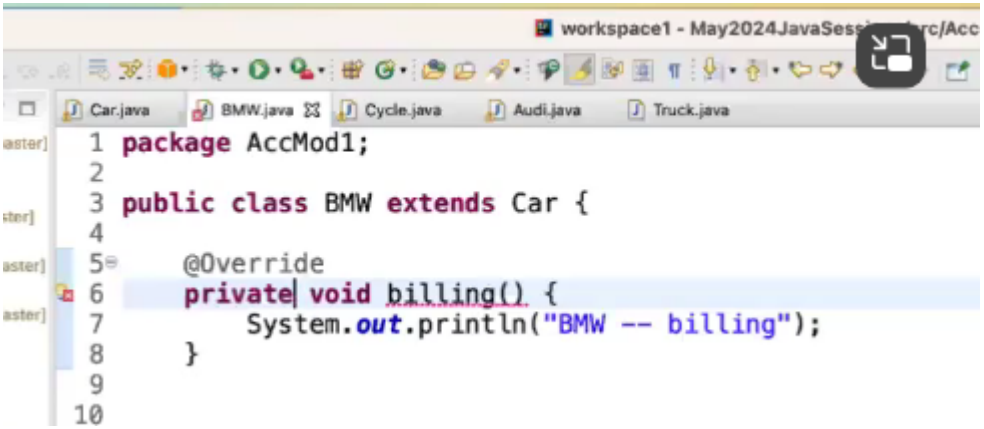


Public allowed-

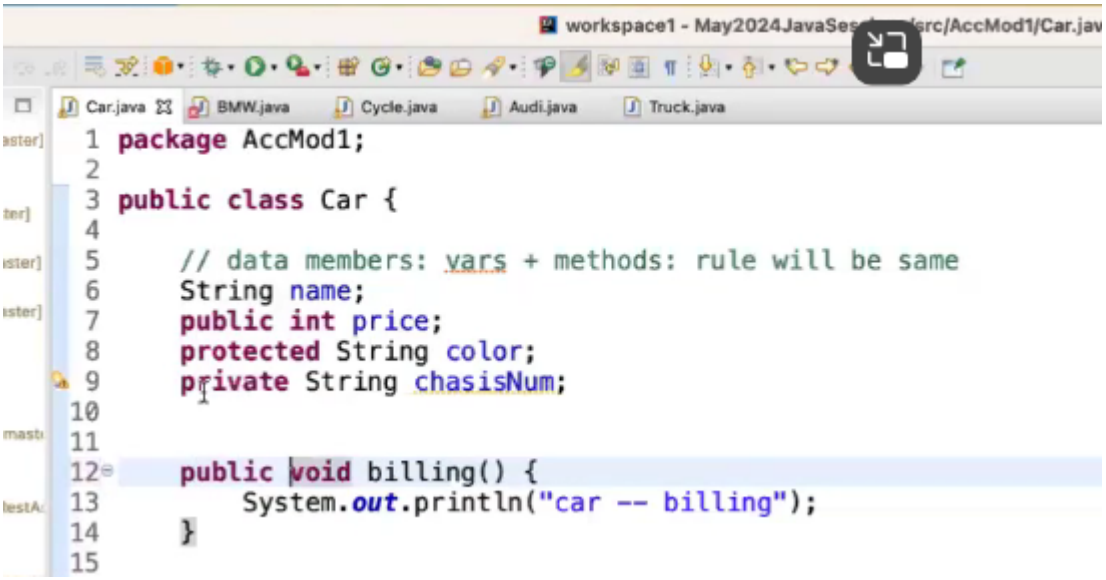


Private – not allowed-

//Cannot reduce the visibility of the inherited method from car2

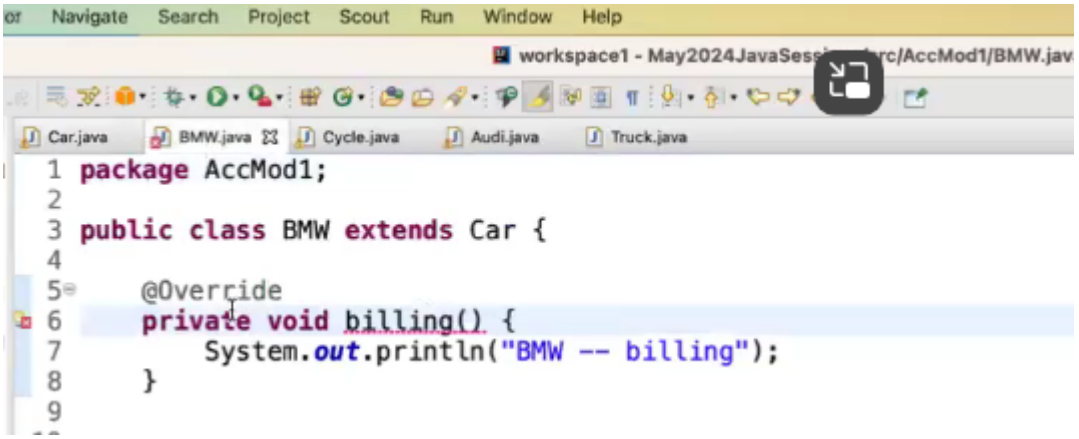


Public in parent-



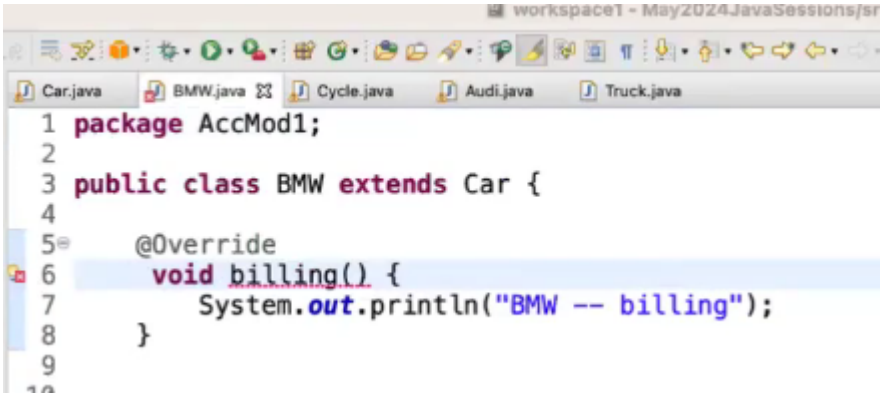
Private not allowed in child-

// Cannot reduce the visibility of the inherited method from car2



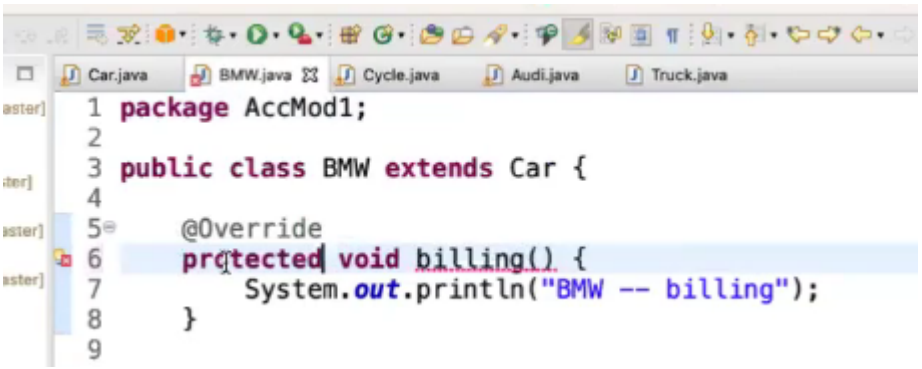
Default not allowed in child-

//Cannot reduce the visibility of the inherited method from car2

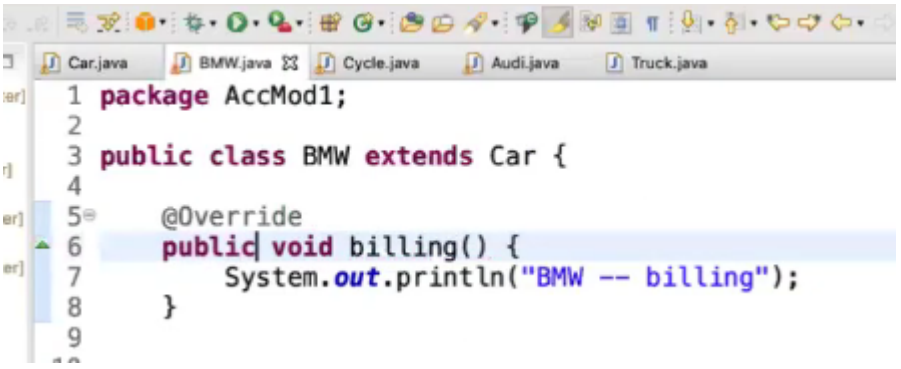


Protected not allowed in child-

//Cannot reduce the visibility of the inherited method from car2



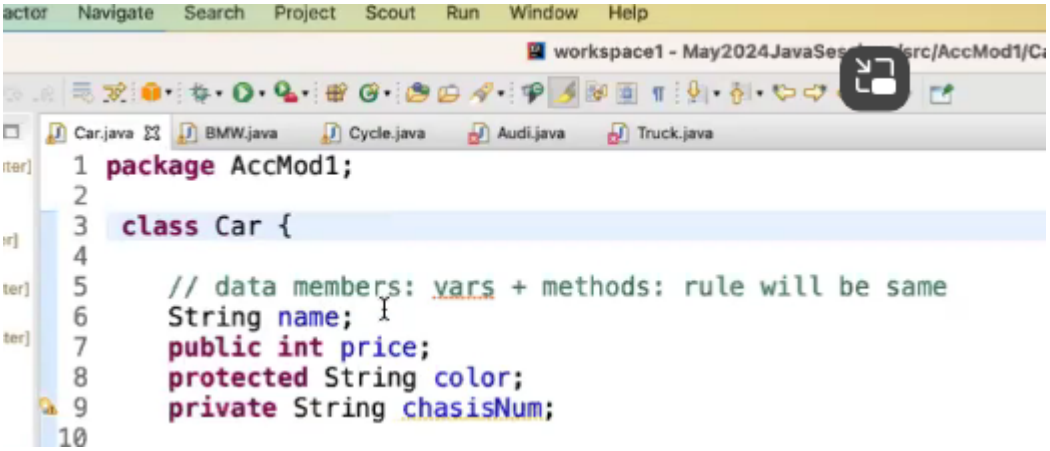
Public allowed in child-



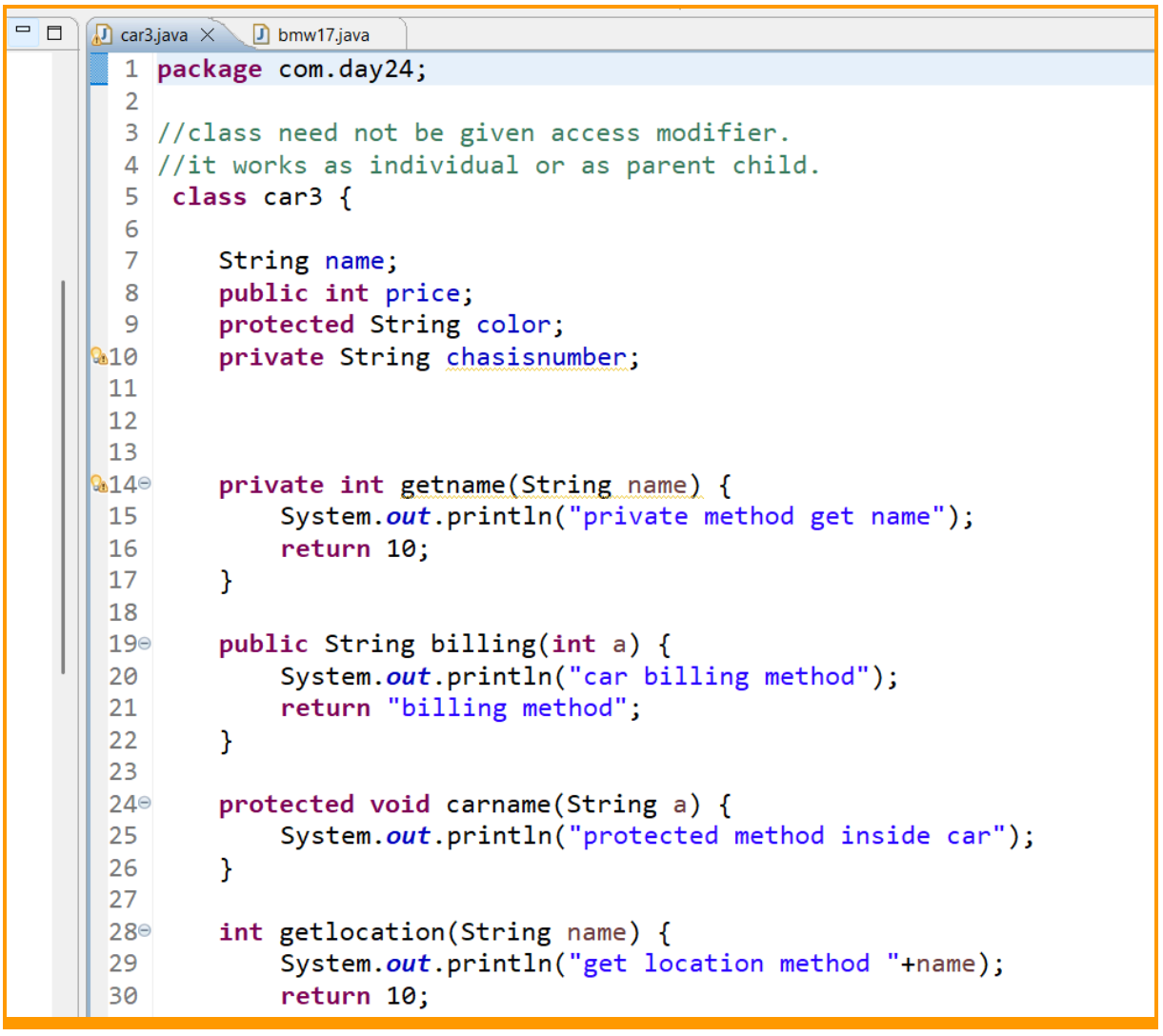
Class can be public, abstract, final only.

Class without any specifier-

Allowed but not a default class.



paste bmw17, car3-



```
29          System.out.println("get location method "+name);
30          return 10;
31      }
32
33  }
34
```

bmw17.java

```
1  package com.day24;
2
3  //class need not be given any access modifier its ok.
4   class bmw17  extends car3{
5
6      public static void main(String[] args) {
7
8
9          bmw17 b1=new bmw17();
10         int s1=b1.price;
11         System.out.println(s1);
12         String s2=b1.billing(10);
13         System.out.println(s2);
14
15     }
16
17     //public can override the public from parent class.
18     @Override
19     public String billing(int a) {
20         System.out.println("child class bmw billing method " +a);
21         return "billing method";
22     }
23
24 }
```

```
23
24 }
25
26 //0
27 //child class bmw billing method 10
28 //billing method
29
```

paste bmw18-

bmw18.java

```java
package com.day24;

//Illegal modifier for the class bmw18; only public, abstract & final are permitted
//class can be public abstract or final only.
 protected class bmw18  extends car3{

    public static void main(String[] args) {


        bmw18 b1=new bmw18();
        int s1=b1.price;
        System.out.println(s1);
        String s2=b1.billing(10);
        System.out.println(s2);

    }

    //public can override the public from parent class.
    @Override
    public String billing(int a) {
        System.out.println("child class bmw billing method " +a);
        return "billing method";
    }

}
```

paste bmw19-

bmw19.java

```java
package com.day24;

//Illegal modifier for the class bmw18; only public, abstract & final are permitted
//class can be public abstract or final only.
 private class bmw19  extends car3{

    public static void main(String[] args) {


        bmw19 b1=new bmw19();
        int s1=b1.price;
        System.out.println(s1);
        String s2=b1.billing(10);
        System.out.println(s2);

    }

    //public can override the public from parent class.
    @Override
    public String billing(int a) {
        System.out.println("child class bmw billing method " +a);
        return "billing method";
    }

}
```

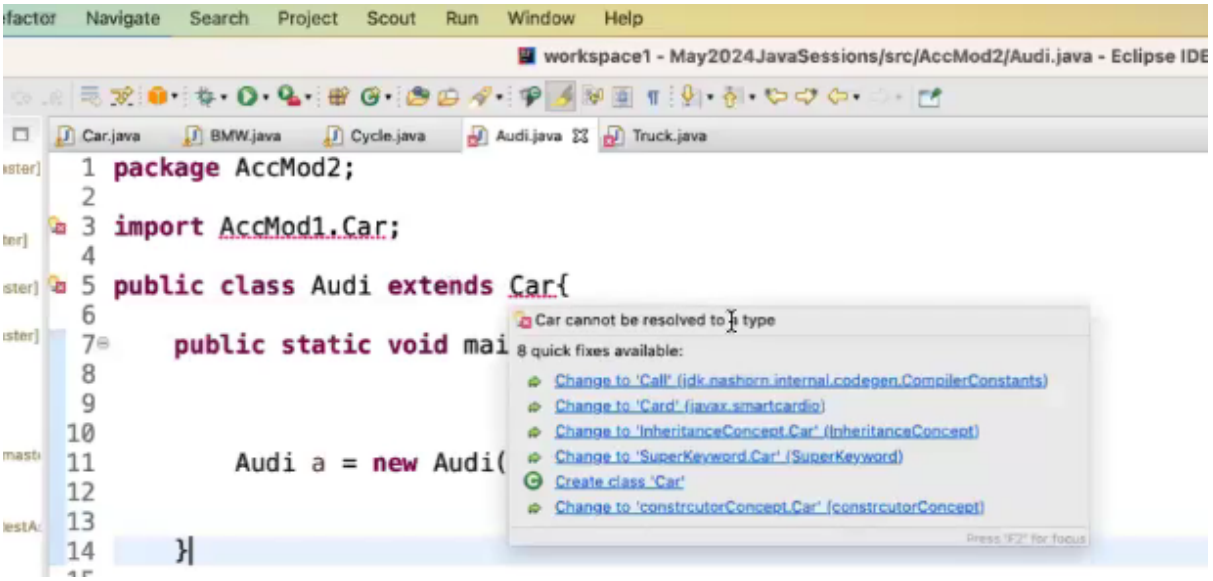
paste bmw20-

bmw20.java

```java
package com.day24;

//Syntax error on token "default", delete this token
//class can be public abstract or final only.
 default class bmw20  extends car3{

    public static void main(String[] args) {


        bmw20 b1=new bmw20();
        int s1=b1.price;
        System.out.println(s1);
        String s2=b1.billing(10);
        System.out.println(s2);

    }

    //public can override the public from parent class.
    @Override
    public String billing(int a) {
        System.out.println("child class bmw billing method " +a);
        return "billing method";
    }

}
```

~~Cannot make child class of the above class~~

now its allowed.



Access modifier for constructor, variables and methods only.

Constructor-

paste car4-

car4.java

```java
package com.day24;

class car4 {

    String name;
    public int price;
    protected String color;
    private String chasisnumber;

    //create constructor public.
    public car4(String name, int price, String color, String chasisnumber) {
        this.name=name;
        this.price=price;
        this.color=color;
        this.chasisnumber=chasisnumber;

        System.out.println(this.name);
        System.out.println(name);
        System.out.println(this.price);
        System.out.println(price);
        System.out.println(this.color);
        System.out.println(color);
        System.out.println(this.chasisnumber);
        System.out.println(chasisnumber);
        System.out.println("parent class constructor");
    }

}
```

paste bmw21-

bmw21.java

```java
package com.day24;


//Implicit super constructor car4() is undefined for
//default constructor. Must define an explicit constructor

/**
 * since parent class has parameter constructor so child
 * class doesnt allow me to proceed without creating
 * parameter constructor.
 */
 public class bmw21  extends car4{

    public static void main(String[] args) {

    //public constructor can be accessed in another class.


}

```

paste bmw22-

bmw22.java

```java
package com.day24;

 public class bmw22  extends car4{

    public bmw22() {
        //default constructor not allowed.
        //since parent has parameter constructor.
        //Implicit super constructor car4()
        //is undefined. Must explicitly invoke another constructor
    }

    public static void main(String[] args) {

    //public constructor can be accessed in another class.


}
```

paste bmw23-

bmw23.java

```java
package com.day24;


 public class bmw23  extends car4{

    public bmw23() {
        super("bmw", 10000, "black", "1234");
        //this is one way to resolve.
    }

    public static void main(String[] args) {

    //public constructor can be accessed in another class.


}
 }

```

paste bmw24-

bmw24.java

```java
package com.day24;


public class bmw24  extends car4{

   //this is another way to resolve.
    //create the same constructor in child and add more variables if needed.
    String address;
    int phone;

   public bmw24(String name, int price, String color, String chasisnumber, String address
           ,int phone) {
       super(name, price, color, chasisnumber);
       System.out.println(this.name);
       System.out.println(name);
       System.out.println(this.price);
       System.out.println(price);
       System.out.println(this.color);
       System.out.println(color);
//       System.out.println(this.chasisnumber);//The field car4.chasisnumber is not visible
       System.out.println(chasisnumber);
       System.out.println("child class constructor");
       System.out.println(this.address);
       System.out.println(address);
       System.out.println(this.phone);
       System.out.println(phone);
   }
```

```
        System.out.println(phone);
    }

    public static void main(String[] args) {

    //public constructor can be accessed in another class.
        car4 c1=new car4("karan", 23434, "green", "@#4324");
        System.out.println(c1);

}
 }


// karan
// karan
// 23434
// 23434
// green
// green
// @#4324
// @#4324
// parent class constructor
// com.day24.car4@24d46ca6
```

paste car5. bmw25-

bmw25.java car5.java

```java
package com.day24;

 class car5 {

    String name;
    public int price;
    protected String color;
    private String chasisnumber;

    //create constructor private.
    private car5(String name, int price, String color, String chasisnumber) {
        this.name=name;
        this.price=price;
        this.color=color;
        this.chasisnumber=chasisnumber;

        System.out.println(this.name);
        System.out.println(name);
        System.out.println(this.price);
        System.out.println(price);
        System.out.println(this.color);
        System.out.println(color);
        System.out.println(this.chasisnumber);
        System.out.println(chasisnumber);
        System.out.println("parent class constructor");
    }

}
```

bmw25.java

```java
package com.day24;


 public class bmw25  extends car5{

    //this is another way to resolve.
     //create the same constructor in child and add more variables if needed.
     String address;
     int phone;

     //Implicit super constructor car5()
     //is undefined. Must explicitly invoke another constructor
    public bmw25(String name, int price, String color, String chasisnumber, String address
            ,int phone) {
        //The constructor car5(String, int, String, String) is not visible
        //private constructor cannot be accessed from outside class.
//      super(name, price, color, chasisnumber);
        System.out.println(this.name);
        System.out.println(name);
        System.out.println(this.price);
        System.out.println(price);
        System.out.println(this.color);
        System.out.println(color);
//      System.out.println(this.chasisnumber);//The field car4.chasisnumber is not visible
        System.out.println(chasisnumber);
        System.out.println("child class constructor");
        System.out.println(this.address);
        System.out.println(address);
        System.out.println(this.phone);
        System.out.println(phone);
```

```
29          System.out.println(this.phone);
30          System.out.println(phone);
31      }
32
33      public static void main(String[] args) {
34
35      //private constructor cannot be accessed in another class.
36          //The constructor car5(String, int, String, String) is not visible
37          car5 c1=new car5("karan", 23434, "green", "@#4324");
38          System.out.println(c1);
39
40  }
41   }
42
```

paste car6, bmw26-

bmw26.java car6.java

```
1  package com.day24;
2
3   class car6 {
4
5      String name;
6      public int price;
7      protected String color;
8      private String chasisnumber;
9
10     //create constructor protected.
11     protected car6(String name, int price, String color, String chasisnumber) {
12         this.name=name;
13         this.price=price;
14         this.color=color;
15         this.chasisnumber=chasisnumber;
16
17         System.out.println(this.name);
18         System.out.println(name);
19         System.out.println(this.price);
20         System.out.println(price);
21         System.out.println(this.color);
22         System.out.println(color);
23         System.out.println(this.chasisnumber);
24         System.out.println(chasisnumber);
25         System.out.println("parent class constructor");
26     }
27
28 }
29
30
```

bmw26.java

```java
package com.day24;


public class bmw26  extends car6{

   //this is another way to resolve.
    //create the same constructor in child and add more variables if needed.
    String address;
    int phone;

    //protected constructor can be accessed in child class.
   public bmw26(String name, int price, String color, String chasisnumber, String address
           ,int phone) {
       super(name, price, color, chasisnumber);
       System.out.println(this.name);
       System.out.println(name);
       System.out.println(this.price);
       System.out.println(price);
       System.out.println(this.color);
       System.out.println(color);
//       System.out.println(this.chasisnumber);//The field car4.chasisnumber is not visible
       System.out.println(chasisnumber);
       System.out.println("child class constructor");
       System.out.println(this.address);
       System.out.println(address);
       System.out.println(this.phone);
       System.out.println(phone);
   }
```

```java
        System.out.println(phone);
    }

    public static void main(String[] args) {

    //protected constructor can be accessed in another class.
        car6 c1=new car6("karan", 23434, "green", "@#4324");
        System.out.println(c1);


}
 }

// karan
// karan
// 23434
// 23434
// green
// green
// @#4324
// @#4324
// parent class constructor
// com.day24.car4@4517d9a3
```

paste car7-

car7.java

```java
package com.day24;

class car7 {

    String name;
    public int price;
    protected String color;
    private String chasisnumber;

    //create constructor default.
    //not allowed -
//  Syntax error on token "car7", Identifier expected after this token
    default car7(String name, int price, String color, String chasisnumber) {
        this.name=name;
        this.price=price;
        this.color=color;
        this.chasisnumber=chasisnumber;

        System.out.println(this.name);
        System.out.println(name);
        System.out.println(this.price);
        System.out.println(price);
        System.out.println(this.color);
        System.out.println(color);
        System.out.println(this.chasisnumber);
        System.out.println(chasisnumber);
        System.out.println("parent class constructor");
    }

}
```

Can be public, private, protected, default.

Protected constructor cannot be accessed in different package child class also-

//car6 cannot be resolved to a type

Car.java | BMW.java | Cycle.java | Audi.java | Truck.java

```java
package AccMod2;

import AccMod1.Car;

public class Audi extends Car{

    public static void main(String[] args) {


        Audi a = new Audi();

        //Car c = new Car();//protected const... can not be accessed from a differnt package parent class

```