1 Number Reverse program

```
package test1;
import java.util.Scanner;
class Revert1{
    Scanner s = new Scanner(System.in);
    int remainder,result=0;
    public void m1() {
        System.out.println("Enter the number:");
        int n=s.nextInt();
        while(n !=0) {
            remainder=n%10;
            result=result*10+remainder;
            n=n/10;
                }
        System.out.print("Reverse Number:"+result);

    }
}
public class Revert {

    public static void main(String[] args) {
        Revert1 r = new Revert1();
        r.m1();

    }

}
```

Output:

```
Enter the number:
2343
Reverse Number:3432
```

2. Pattern program:

```
package collection;

public class Pattern1 {

        public static void main(String[] args) {
            printAsteriskPattern();
        }

        private static void printAsteriskPattern() {
            int[] pattern = {1, 2, 4, 2, 5, 1, 4};

            for (int i : pattern) {
                for (int j = 0; j < i; j++) {
                    System.out.print("*");
                }
                System.out.println();
            }
        }
    }
```

Output:

3 using try catch block If person is above 18 are eligible to vaccine else not eligible to vaccine using exception throws

```
package test1;
```

```java
import java.util.Scanner;
import java.util.*;
public class Try {

    static int n=0;
        public static void main(String[] args) {
                try(Scanner s = new Scanner(System.in);){//try
with resource
                        System.out.print("Enter  age:");
                        n=s.nextInt();
                }
        try{
                if(n>=18) {
                        System.out.println("Eligible to vaccine.");
                }
                else {
                        System.out.println("Not eligible to vaccine.");
                }

        }
        catch(Exception n) {
                System.out.println("It is negative value.so enter
positive value.");
        }

        }

}
```

Output:

```
Enter  age:20
Eligible to vaccine.
```

4 Using Arraylist find the highest salary of employee .Example (ram,HR,10000)

```
package test1;
import java.util.ArrayList;

        class data {
            private String name;
            private String position;
            private long salary;

            public data(String name, String position, long salary)
{
                this.name = name;
                this.position =position;
                this.salary = salary;
            }

            public String getName() {
                return name;
            }

            public String getPosition() {
                return position;
            }

            public long getSalary() {
                return salary;
            }

            @Override
            public String toString() {
                return "Employee{" + "name='" + name + '\'' +
                        ", position='" + position + '\'' +
                        ", salary=" + salary +
                        '}';
            }
        }
        public class Employ {

            public static void main(String[] args) {
                ArrayList<data> employeeList = new
ArrayList<>();

                // Create employee instances
                data employee1 = new data("Ilakkiya", "HR",
19000);
                data employee2 = new data("Vignesh", "Testing",
15000);
                data employee3 = new data("Pushparaj", "Finance",
12000);
                data employee4 = new data("Keerathana",
"Developer", 14000);

                // Add employees to the ArrayList
                employeeList.add(employee1);
                employeeList.add(employee2);
                employeeList.add(employee3);

                // Find the employee with the highest salary
                data highest = high(employeeList);
                if (highest != null) {
                    System.out.println("Employee with highest
salary: " + highest);
                } else {
                    System.out.println("No employees found.");
                }
            }

            public static data high(ArrayList<data>
employeeList) {
                if (employeeList.isEmpty()) {
                    return null;
                }

                data highestPaidEmployee = employeeList.get(0);
                long maxSalary =
highestPaidEmployee.getSalary();

                for (data employee : employeeList) {
                    if (employee.getSalary() > maxSalary) {
                        maxSalary = employee.getSalary();
                        highestPaidEmployee = employee;
                    }
                }

                return highestPaidEmployee;
            }
        }
```

Ouptu:

```
Employee with highest salary: Employee{name='Ilakkiya',
position='HR', salary=19000}
```

5 Find the following data

```
package collection;
import java.util.*;

public class Mapp {

   public static void main(String[] args) {
       Map<String, ArrayList<String>> place = new
HashMap<>();
       place.put("Ram", new
ArrayList<>(Arrays.asList("Chennai", "Adyar")));
       place.put("Deva", new
ArrayList<>(Arrays.asList("Chennai", "Adyar")));
       place.put("Siva", new
ArrayList<>(Arrays.asList("Chennai", "Vadapalani")));
       place.put("Kishore", new
ArrayList<>(Arrays.asList("Kerala", "Thrissur")));
       place.put("Ganesh", new
ArrayList<>(Arrays.asList("Bangalore", "WhiteField")));
       place.put("Shivan", new
ArrayList<>(Arrays.asList("Chennai", "Chormpet")));

       displayOutput(place);
   }

   private static void displayOutput(Map<String,
ArrayList<String>> place) {
       Map<String, Integer> locationCount = new
HashMap<>();

       for (Map.Entry<String, ArrayList<String>> entry :
place.entrySet()) {
           String location = entry.getValue().get(0) + "," +
entry.getValue().get(1);
```

```
            locationCount.put(location,
locationCount.getOrDefault(location, 0) + 1);
        }
        for (Map.Entry<String, Integer> entry :
locationCount.entrySet()) {
            System.out.println(entry.getKey() + " - " +
entry.getValue());
        }
    }
}
```

Output:

```
Kerala,Thrissur - 1
Chennai,Chormpet - 1
Chennai,Vadapalani - 1
Chennai,Adyar - 2
Bangalore,WhiteField - 1
```