

Q1:

```
public class Divisors {  
    public static void main (String[] args) {  
        /// Gets input from user as int  
        int num = Integer.parseInt(args[0]);  
        /// Starts checking every number from 1 to num that was given by user  
        for (int i = 1; i < num+1; i++) {  
            /// if num % i returns true, num % i is a divisor of num  
            /// therefore, we will save it under string 'result'  
            if (num % i == 0) {  
                System.out.println(i);  
            }  
        }  
    }  
}
```

Q2:

```
public class Reverse {  
    public static void main (String[] args){  
        /// Gets input from user as string  
        String input = args[0];  
        for (int i = 0; i < input.length(); i++ ) {  
            int idx = input.length() - 1 - i;  
            System.out.print(input.charAt(idx));  
        }  
        System.out.println();  
        /// Calculates the length of the string to find it's middle character  
        /// if the length is an even number, it returns the character at index (len \ 2)  
        /// if the length is an odd number, it returns the character at rounded up index (len + 1 \ 2)  
        if (input.length() % 2 == 0) {  
            int idx = ( input.length() ) / 2 - 1;  
            System.out.println("The middle character is " + input.charAt(idx));  
        } else {  
            int idx = (input.length() ) / 2;  
            System.out.println("The middle character is " + input.charAt(idx));  
        }  
    }  
}
```

Q3:

```
public class InOrder {  
    public static void main (String[] args) {  
        /// Generate a number in range [0,10) and generate a boolean to check if decreasing  
        order value is met  
  
        int num = (int) (Math.random() * 10);  
  
        boolean isDecreasing = false;  
  
        /// As long as num - new_num != 1, the loop will generate a new number and check if  
        criteria is met  
  
        do {  
            int new_num = (int) (Math.random() * 10);  
  
            System.out.print(num + " ");  
  
            if (num - new_num >= 1) {  
                isDecreasing = true;  
            }  
  
            num = new_num;  
        }  
  
        while (isDecreasing == false);  
    }  
}
```

Q4:

```
public class Perfect {  
    public static void main (String[] args) {  
        /// Gets input from user as int  
        int num = Integer.parseInt(args[0]);  
        /// Initiate the answer string as well as the sum value  
        int sum = 1;  
        String answer = (num + " is a perfect number since " + num + " = 1");  
        for (int i = 2; i < num; i++) {  
            /// For each i checks if i is a divisor of num, and if so adds it to the sum &  
            /// concatenates to the answer  
            if (num % i == 0) {  
                answer = answer + (" + ") + i;  
                sum = sum + i;  
            }  
        }  
        /// Checks if the sum calculated in the loop is equal to the input number  
        /// If yes, then we have a perfect number; Otherwise, it is not a perfect number  
        if (num == sum) {  
            System.out.println(answer);  
        } else {  
            System.out.println(num + " is not a perfect number");  
        }  
    }  
}
```

Q5:

```
public class DamkaBoard {  
    public static void main(String[] args) {  
        /// Gets input from user  
        int num = Integer.parseInt(args[0]);  
        /// Creates a line in the damka board based on the number given by user  
        String base = (" *");  
        String line_even = (" *") + base.repeat(num - 1);  
        String line_odd = ("*") + base.repeat(num - 1) + (" ");  
        for (int i = 1; i <= num; i++) {  
            /// For each row checks if it's even or odd.  
            /// odd --> prints line as is; even --> adds space infront of the line to create  
            indentation  
            if ( i % 2 == 1) {  
                System.out.println(line_odd);  
            } else {  
                System.out.println(line_even);  
            }  
        }  
    }  
}
```

Q6:

```
public class OneOfEach {  
    public static void main (String[] args) {  
          
        //// Generate a boolean argument to follow if the family has a boy / girl and a count  
        integer  
  
        boolean girl = false;  
  
        boolean boy = false;  
  
        int count = 0;  
  
        while (girl == false || boy == false) {  
              
            /// Generate a random number between [0, 1)  
  
            /// values greater than 0.5 are considered girls, lower are considered boys  
  
            double number = Math.random();  
  
            if (number > 0.5) {  
                girl = true;  
  
                System.out.print("g ");  
  
            } else {  
                boy = true;  
  
                System.out.print("b ");  
  
            }  
  
            count++;  
  
        }  
  
        System.out.println();  
  
        System.out.println("You made it... and you now have " + count + " children.");  
    }  
}
```

Q7:

```
public class OneOfEachStats1 {  
    public static void main (String[] args) {  
        /// Accepts input from user as integer  
        int T = Integer.parseInt(args[0]);  
        int sum = 0;  
        int twoChildren = 0;  
        int threeChildren = 0;  
        int fourChildren = 0;  
        for (int i = 1; i <= T; i++) {  
            boolean girl = false;  
            boolean boy = false;  
            int count = 0;  
            while (girl == false || boy == false) {  
                /// Generate a random number between [0, 1)  
                /// values greater than 0.5 are considered girls, lower are considered boys  
                double number = Math.random();  
                if (number > 0.5) {  
                    girl = true;  
                    //System.out.print("g ");  
                } else {  
                    boy = true;  
                    //System.out.print("b ");  
                }  
                count++;  
            }  
            //System.out.println();  
            //System.out.println("You made it... and you now have " + count + " children.");  
            sum = sum + count;  
        }  
    }  
}
```

```

        if (count == 2) {
            twoChildren++;
        } else if (count == 3) {
            threeChildren++;
        } else if (count >= 4) {
            fourChildren++;
        }
    }

    double avg = (double) sum / T ;
    System.out.println("Average: " + avg + " children to get at least one of each gender.");
    System.out.println("Number of families with 2 children: " + twoChildren);
    System.out.println("Number of families with 3 children: " + threeChildren);
    System.out.println("Number of families with 4 or more children: " + fourChildren);
    int max = Math.max(threeChildren, fourChildren);
    max = Math.max(max, twoChildren);
    if (max == twoChildren) {
        System.out.println("The most common number of children is 2.");
    } else if (max == threeChildren) {
        System.out.println("The most common number of children is 3.");
    } else {
        System.out.println("The most common number of children is 4 or more.");
    }
}
}

```



Q8:

```
import java.util.Random;
```

```
public class OneOfEachStats {
```

```
    public static void main (String[] args) {
```

```
        // Gets the two command-line arguments
```

```
        int T = Integer.parseInt(args[0]);
```

```
        int seed = Integer.parseInt(args[1]);
```

```
        // Initailizes a random numbers generator with the given seed value
```

```
        Random generator = new Random(seed);
```

```
        int sum = 0;
```

```
        int twoChildren = 0;
```

```
        int threeChildren = 0;
```

```
        int fourChildren = 0;
```

```
        for (int i = 1; i <= T; i++) {
```

```
            boolean girl = false;
```

```
            boolean boy = false;
```

```
            int count = 0;
```

```
            while (girl == false || boy == false) {
```

```
                /// Generate a random number between [0, 1)
```

```
                /// values greater than 0.5 are considered girls, lower are considered boys
```

```
                double number = generator.nextDouble();
```

```
                if (number > 0.5) {
```

```
                    girl = true;
```

```
                    //System.out.print("g ");
```

```
                } else {
```

```
                    boy = true;
```

```
                    //System.out.print("b ");
```

```
                }
```

```
            count++;
```

```

    }

    //System.out.println();

    //System.out.println("You made it... and you now have " + count + " children.");

    sum = sum + count;

    if (count == 2) {
        twoChildren++;
    } else if (count == 3) {
        threeChildren++;
    } else if (count >= 4) {
        fourChildren++;
    }
}

double avg = (double) sum / T ;

System.out.println("Average: " + avg + " children to get at least one of each gender.");

System.out.println("Number of families with 2 children: " + twoChildren);

System.out.println("Number of families with 3 children: " + threeChildren);

System.out.println("Number of families with 4 or more children: " + fourChildren);

int max = Math.max(threeChildren, fourChildren);

max = Math.max(max, twoChildren);

if (max == twoChildren) {
    System.out.println("The most common number of children is 2.");
} else if (max == threeChildren) {
    System.out.println("The most common number of children is 3.");
} else {
    System.out.println("The most common number of children is 4 or more.");
}

}

}

```