L31

Sunday, January 30, 2022 10:37 PM

Saurav Hathi

https://www.youtube.com/channel/UCp6MFWao5vWRnyRCxBsKnfw

Problem solving – Text processing

Q1. Write a class Greeting with a **main** method. The method receives one command line argument and prints the output as shown below.

For example:
Cmd Args : Ram
Hello Ram!
Note: Make sure to use println and not print method. If you want to use printf, ensure to print a \n character at the end.

```
package q11191;

public class Greeting{
    public static void main(String[] args){
        System.out.println("Hello "+args[0]+"!");
    }
}
```

Q2. Write a class PrefixCheck with a **main** method. The method receives one command line argument. If the argument starts with **en** print it as it is, else prepend **en** and print the output.

These are examples for your understanding :
Cmd Args : bye
enbye
Cmd Args : entertainment
entertainment

```
package q11192;

public class PrefixCheck{
    public static void main(String[] args){
        if(args[0].startsWith("en")){
        System.out.println(args[0]);
        } else{
            System.out.println("en"+args[0]);
        }
    }
}
```

Q3. Write a class ExpandAString with a **main** method. The program receives two command line arguments. The first argument is a string and the second argument is a number. Print the first argument as many number of times as the second argument.

For Example:
Cmd Args : Ganga 3
GangaGangaGanga
Note: Make sure to use println and not print method.

```
package q11193;

public class ExpandAString {
    public static void main(String[] args){

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

    System.out.println(args[0].repeat(a)):

    }
}
```

String parsing exercise

Q4. Write a class TestSubstring with a **main** method. The method receives one command line argument and print the output formed by the first three characters of the argument.

For example:

Cmd Args : Important
Imp

```java
package q11194;

public class TestSubstring{
    public static void main(String[] args){

        System.out.println(args[0].substring(0,3));
    }
}
```

Q5. Write a class TestSubstring with a **main** method. The method receives one command line argument and print the **first half of** the argument.

Here is an example:
Cmd Args : Eight
Ei

```java
package q11195;
public class TestSubstring {

    public static void main(String[] args){

        int size = (args[0].length());

        System.out.println(args[0].substring(0,size/2));
    }

}
```

Q6. Write a class TestSubstring with a **main** method. The method receives one command line argument and prints the output by removing the first and last characters of the argument.

For example:
Cmd Args : January
anuar

```java
package q11196;

public class TestSubstring{
    public static void main(String[] args){

        System.out.println(args[0].substring(1,args[0].length()-1));
    }
}
```

Q7. Write a class StringStairs with a **main** method. The method receives one command line argument and prints the first char in one line, then the first two chars in the next line, etc.

For example:
Cmd Args : Delhi
D
De
Del
Delh
Delhi

```java
package q11197;

public class StringStairs{
    public static void main(String[] args){

        for(int i = 1; i < args[0].length()+1; i++){

            System.out.println(args[0].substring(0,i));
        }
    }
}
```

Q8. Write a class RemoveChar with a **main** method. The method receives one command line argument and prints the output by removing the **3**rd character from the argument.

Assumptions:
1. Length of the argument is greater than or equal to 3 characters

For example:
Cmd Args : Victory

```java
package q11198;

public class RemoveChar{
    public static void main(String[] args){

        int size = args[0].length();

        System.out.println(args[0].substring(0,2)+args[0].substring(3,size));
    }
}
```

Q9. Write a class ExtractTag with a **main** method. The method receives one command line argument in between tags like **[ ]**, extract the argument between the tags and print the output.

For Example:
Cmd Args : [Independent]
Independent

```java
package q11199;

public class ExtractTag{
    public static void main(String[] args){

        System.out.println(args[0].replace("[","").replace("]",""));
    }
}
```

Q10. Write a class AttachTag with a **main** mehod. The method receives two command line arguments make the **second argument** as a special tag and attach before and after the **first argument** and print the value.

For example:
Cmd Args : Delhi New
[New]Delhi[/New]
Note: Make sure to use println and not print method.

```java
package q11200;

public class AttachTag{
    public static void main(String[] args){
        String s1 = "["+args[1]+"]";
        String s2 = "[/"+args[1]+"]";
        String a = s1+args[0]+s2;
        System.out.println(a);
    }
}
```

Q11. Write a class PutInsideTag with a **main** method. The method receives two command line arguments. First one is a tag like '[]' or '{{{}}}' etc. The second argument is a string. Write logic to print a new word where the second argument is kept exactly in the middle of the tag.

For example:
Cmd Args : {{{}}} Hyderabad
{{{{Hyderabad}}}}
Note: Make sure to use println and not print method.

```java
package q11201;

public class PutInsideTag{
    public static void main(String[] args){

        int mid = args[0].length()/2;

        String s1 = args[0].substring(0,mid);

        String s2 = args[0].substring(mid,args[0].length());

            System.out.println(s1+args[1]+s2);
    }
}
```

Q12. Write a class ReverseString with a **main** method. The method receives one command line argument. Write a logic to **reverse** it and print the output.

For Example:
Cmd Args : Ram lakshman
namhskal maR

```java
package q11202;

public class ReverseString{
    public static void main(String[] args){

        for(int i = 0; i < args[0].length(); i++){

            System.out.print(args[0].charAt(args[0].length()-1-i));
        }

        System.out.println("\n");
    }
}
```

Q13. Write a class MiddleTwoChars with a **main** method. The method receives one command line argument extract the **middle two characters** from the argument and print the output.

Assumptions:
1. The string has even number of characters

For example:
Cmd Args : java
av
Note: Make sure to use println and not print method.

```java
package q11203;

public class MiddleTwoChars {

    public static void main(String[] args){

        String result = args[0].substring(args[0].length()/2-1, args[0].length()/2+1);

        System.out.println(result);
    }

}
```

Q14. Write a class EndsWith with a **main** method. The method receives one command line argument. Print **true** if the argument ends with **bad**, else print **false**.

For Example:
Cmd Args : Hyderabad
true
Note: Make sure to use println and not print method.

```java
package q11204;

public class EndsWith{
    public static void main(String[] args){

        boolean ends = args[0].endsWith("bad");

        if(ends){

            System.out.println(ends);

        } else{

            System.out.println(ends);

        }
    }
}
```

Q15. Write a class EndsWith with a **main** method. The method receives two command line arguments. Print **true** if the first argument ends with the second argument (ignoring case), else print **false**.

Example:
Cmd Args : Godavari ri
true

[**Hint:** Since we have to verify ignoring the case, you may want to first convert both either to lower or upper case before comparing.]

```java
package q11205;

public class EndsWith {

    public static void main(String[] args){


        boolean ends = args[0].toLowerCase().endsWith(args[1].toLowerCase());

        System.out.println(ends ? ends : ends);
    }
}
```

Q16. Write a class TestString with a **main** method. The method receives one command line argument. Create a new string by concatenating the **first three** and **last three** characters of the argument and print the output.

Assumptions:
1. String length is at least **six**

For Example:
Cmd Args : Hyderabad
Hydbad

```java
package q11206;

public class TestString{

    public static void main(String[] args){

        String input = args[0];

        int size = input.length();

        System.out.println(input.substring(0,3)+input.substring(size-3, size));
    }
}
```

Q17. Write a class TestString with a **main** method. The method receives one command line argument. Create a string by extracting the **first** character and appending it to the **end** of the argument , then append **aa** at the end.

For example:
Cmd Args : Europe
uropeEaa
Note: Make sure to use println and not print method.

```java
package q11207;

public class TestString{
    public static void main(String[] args){

        String input = args[0];

        int size = input.length();

        String first = input.substring(0,1);

        System.out.println(input.substring(1,size)+first+"aa");
    }
}
```

Q18. Write a class MiddleChar with a **main** method. The method receives one command line argument. Print the **middle** character of the argument.

Assumptions:
1. The length of the string is odd

Example:
Cmd Args : Three
r
Note: Make sure to use println and not print method.

```java
package q11208;

public class MiddleChar{
    public static void main(String[] args){

        String input = args[0];

        if(input.length()%2==0){

            System.out.println(input.charAt(input.length()/2-1));

        } else{

            System.out.println(input.charAt(input.length()/2));

        }
    }
}

```

Q19. Write a class CheckChars with a **main** method. The method receives one command line argument and prints **true** if the argument has the characters **w**, **e** and **b**.

For Example:
Cmd Args : westbengal
true

```java
package q11209;

public class CheckChars{

    public static void main(String[] args){

        String input = args[0];

        System.out.println(input.contains("w") && input.contains("e") && input.contains("b"));
    }
}

```

Q20. Write a class AppendChars with a **main** method. The method receives one commmand line argument. If the argument length is less than 10, append '*' at the end to make the length 10.

For example:
Cmd Args : Smile
Smile*****

```java
package q11210;

public class AppendChars {

    public static void main(String[] args){

        String input = args[0];

        if(input.length() < 10){

            for(int i = input.length(); i < 10; i++){

               input += "*";
            }

            System.out.println(input);

            }

            else{

               System.out.println(input);

            }
    }

}

```

Q21. Write a class RemoveChars with a **main** method. The method receives one command line argument and prints the output by removing all **x** characters from the argument.

For example:
Cmd Args : prefix
prefi

Note: Make sure to use println and not print method.

```java
package q11211;

public class RemoveChars{
    public static void main(String[] args){

        System.out.println(args[0].replace("x",""));
    }
}
```

Q22. Write a class RemoveSuffix with a **main** method. The method receives one command line argument. If the argument has the **same prefix** and **suffixes** up to 3 characters, remove the **suffix** and print the argument.

Example:
Cmd Args : systemsys
system

```java
package q11212;

public class RemoveSuffix {

    public static void main(String[] args){

        String input = args[0];

        int size = input.length();

        if(input.substring(0,3).equals(input.substring(size-3, size))){

            System.out.println(input.substring(0,size-3));

        } else{

            System.out.println(input);

        }
    }
}
```

Q23. *Write a program in a class RemovePrefix with a **main** method. The program receives one command line argument.

The program should remove the **first two characters** from the argument and print the output, except in one condition. The program should skip removal of **x** or **y** if it encounters them in the first two positions.

These are some examples for your understanding:

Sample Input Output 1
Cmd Args : xyz
xyz

Sample Input Output 2
Cmd Args : yXz
yz

Sample Input Output 3
Cmd Args : xygoogle
xygoogle

Sample Input Output 4
Cmd Args : abTree
Tree

Sample Input Output 5
Cmd Args : ayFlower
yFlower

Note: Make sure to use println and not print method.

```java
package q11213;

public class RemovePrefix {

    public static void main(String[] args){

        StringBuffer a = new StringBuffer(args[0]);

           if((a.charAt(0) == 'x') == true && (a.charAt(1) == 'y') == true){

            System.out.println(a);

           } else if(a.charAt(0) != 'y' && a.charAt(1) != 'y' && a.charAt(0) != 'x' && a.charAt(1) != 'x'){

            System.out.println(a.delete(0,2));

           } else if((a.charAt(0) == 'y') == true && (a.charAt(1) == 'x') == true){

            System.out.println(a);

           } else if((a.charAt(0) == 'y') == true && a.charAt(1) != 'x'){

            System.out.println(a.delete(1,2));

           } else if((a.charAt(0) == 'x') == true && a.charAt(1) != 'y'){

            System.out.println(a.delete(1,2));

           } else if(a.charAt(0) != 'y' && a.charAt(0) != 'x'){

            System.out.println(a.delete(0,1));

           } else if(a.charAt(1) != 'y' && a.charAt(1) != 'x'){

            System.out.println(a.delete(1,2));

           }

    }

}
```

Q24. Write a class CountChars with a **main** method. The method receives one command line argument and prints the number of **o's** (it is not zero, it is the alphabet '**o**') present in the argument.

Example:
Cmd Args : ozone
2
Note: Make sure to use println and not print method.

```java
package q11214;

public class CountChars{

    public static void main(String[] args){

        String input = args[0];
        int count = 0;

        for(int i = 0; i < input.length(); i++){

            if(input.charAt(i) == 'o'){
                count++;
            }
        }

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

Q25. Write a Java program to Delete and Remove characters using StringBuffer class

Write a class JavaStringBufferDelete with a **main** method to delete characters from a string using StringBuffer class.

Follow the given instructions.

1. Consider a string "Hello India" and delete **0** to **6** characters in that and print the result.
2. Consider another string "Hello World", delete characters from position **0** to **length** of the entire string and print the result.
3. Consider another string "Hello Java", remove **0**th character and then print the result.

```java
package q11215;

public class JavaStringBufferDelete{

    public static void main(String[] args){

        StringBuffer buf = new StringBuffer("Hello India");

        System.out.println(buf.delete(0,6)+"\n");

        buf = new StringBuffer("Hello Java");

        System.out.println(buf.delete(0,1));

    }
}

```

Q26. Given :

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

String str = "null";
        if (str == null) {
                System.out.println("null");
        } else (str.length() == 0) {
                System.out.println("zero");
        } else {
                System.out.println("some");
        }
}
```

- [ ] null
- [ ] zero
- [ ] some
- [x] Compilation fails.
- [ ] An exception is thrown at runtime.