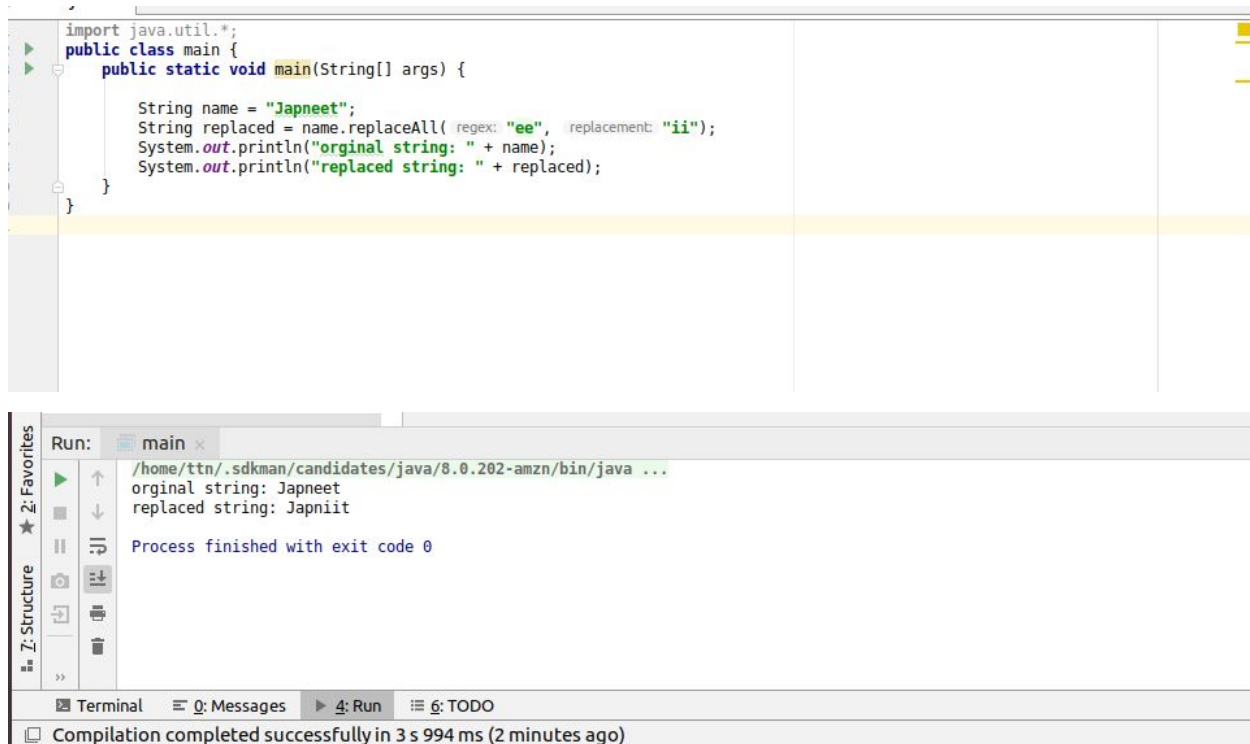


Japneet Kaur (Group-5)

Q1. Write a program to replace a substring inside a string with other string ?



```
import java.util.*;
public class main {
    public static void main(String[] args) {
        String name = "Japneet";
        String replaced = name.replaceAll( regex: "ee", replacement: "ii");
        System.out.println("original string: " + name);
        System.out.println("replaced string: " + replaced);
    }
}
```

Run: main x

```
/home/ttn/.sdkman/candidates/java/8.0.202-amzn/bin/java ...
original string: Japneet
replaced string: Japniit

Process finished with exit code 0
```

Terminal Messages Run TODO

Compilation completed successfully in 3 s 994 ms (2 minutes ago)

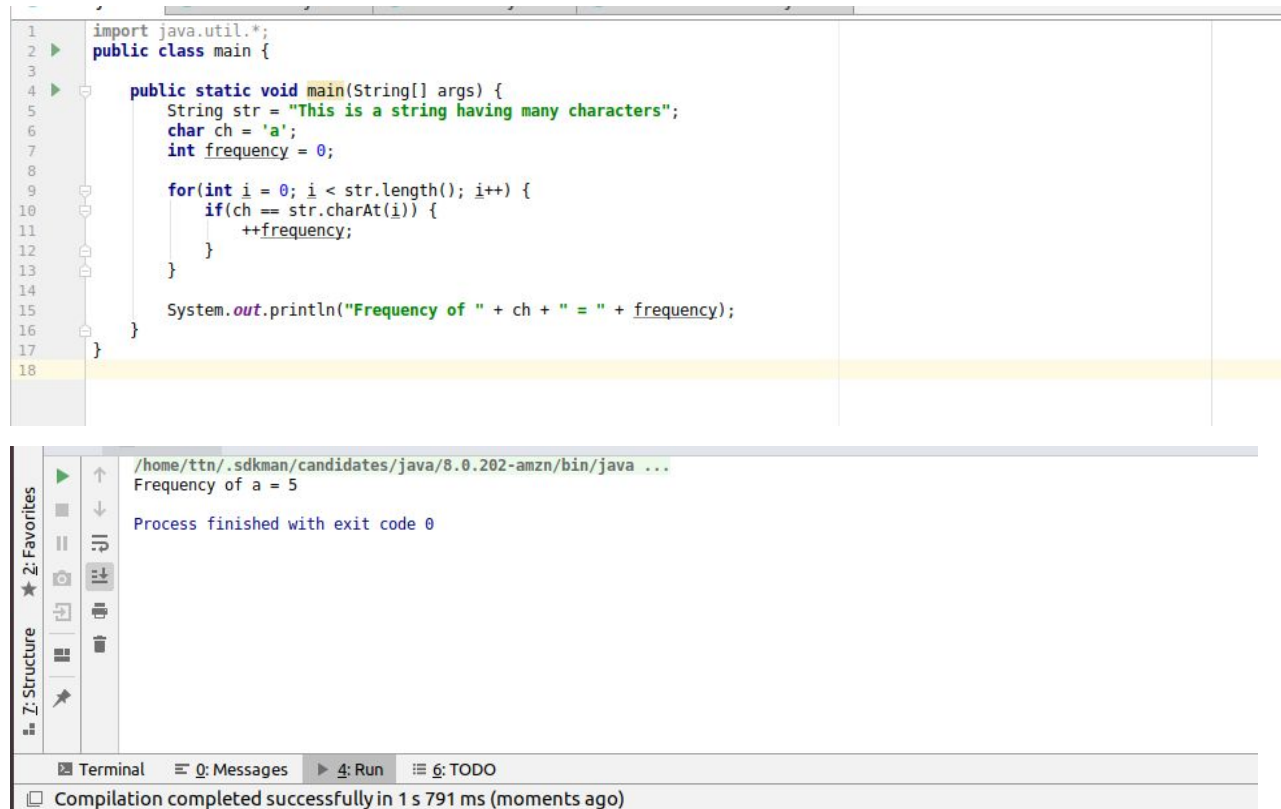
Q2. Write a program to find the number of occurrences of the duplicate words in a string and print them ?

The image shows a screenshot of an IDE with two panels. The top panel displays the source code for a Java class named `CountWords`. The code defines a `main` method that takes an array of strings as input. It splits the input string "This is is a string string one string" into an array of words. It then uses a nested loop to find duplicate words. For each word at index `i`, it compares it with words at indices `j` from `i+1` to the end of the array. If a duplicate is found, it increments a counter `wrc` and sets the word at index `j` to an empty string. Finally, it prints out the words and their counts: "is-2" and "string-3".

```
1 public class CountWords {
2     public static void main(String[] args) {
3
4         String input = "This is is a string string one string";
5         String words[] = input.split(" ");
6         System.out.println("The duplicate words are: ");
7
8         for (int i = 0; i < words.length; i++) {
9             int wrc=1;
10            for (int j = i + 1; j < words.length; j++) {
11                if (words[i].equals(words[j])) {
12                    wrc = wrc + 1;
13                    words[j] = "";
14                }
15            }
16
17            if (words[i] != "" && wrc>1) {
18                System.out.println(words[i] + "-" + wrc);
19            }
20        }
21    }
22 }
23
24
```

The bottom panel shows the IDE's Run console. It displays the command used to run the program, the output "The duplicate words are: is-2 string-3", and a message indicating the process finished successfully with exit code 0. The IDE's status bar at the bottom shows "Compilation completed successfully in 1 s 793 ms (moments ago)".

Q3. Write a program to find the number of occurrences of a character in a string without using loop?



The image shows a screenshot of an IDE with two panels. The top panel displays a Java program that calculates the frequency of the character 'a' in a given string. The code is as follows:

```
1 import java.util.*;
2 public class main {
3
4     public static void main(String[] args) {
5         String str = "This is a string having many characters";
6         char ch = 'a';
7         int frequency = 0;
8
9         for(int i = 0; i < str.length(); i++) {
10             if(ch == str.charAt(i)) {
11                 ++frequency;
12             }
13         }
14
15         System.out.println("Frequency of " + ch + " = " + frequency);
16     }
17 }
18
```

The bottom panel shows the execution output in a terminal window. The command executed is `/home/ttn/.sdkman/candidates/java/8.0.202-amzn/bin/java ...`, and the output is `Frequency of a = 5`. Below the output, it states `Process finished with exit code 0`. The IDE's status bar at the bottom indicates that the compilation was completed successfully in 1 s 791 ms (moments ago).

Q4. Calculate the number & Percentage Of Lowercase Letters,Uppercase Letters, Digits And Other Special Characters In A String

```

1  import java.util.*;
2  public class main {
3
4      public static void main(String args[]) {
5          String data = "Hello HOW are you MR 51";
6          char [] charArray = data.toCharArray();
7          int upper = 0;
8          int lower = 0;
9          int digit = 0;
10         int others = 0;
11
12         int totalChars = data.length();
13         for(int i=0; i<data.length(); i++) {
14             if (Character.isUpperCase(charArray[i])) {
15                 upper++;
16             } else if (Character.isLowerCase(charArray[i])) {
17                 lower++;
18             } else if (Character.isDigit(charArray[i])) {
19                 digit++;
20             } else {
21                 others++;
22             }
23         }
24         System.out.println("Total length of the string :"+totalChars);
25         System.out.println("Upper case :"+upper);
26         System.out.println("Percentage of upper case letters: "+(upper*100)/totalChars+"%");
27         System.out.println("Lower case :"+lower);
28         System.out.println("Percentage of lower case letters: "+(lower*100)/totalChars+"%");
29         System.out.println("Digit :"+digit);
30         System.out.println("Percentage of digits :"+(digit*100)/totalChars+"%");
31         System.out.println("Others :"+others);
32         System.out.println("Percentage of other characters :"+(others*100)/totalChars+"%");
33     }
34 }
35

```

```

/home/ttn/.sdkman/candidates/java/8.0.202-amzn/bin/java ...
Total length of the string :23
Upper case :6
Percentage of upper case letters: 26%
Lower case :10
Percentage of lower case letters:43%
Digit :2
Percentage of digits :8%
Others :5
Percentage of other characters :21%

Process finished with exit code 0

```

Terminal 0: Messages 4: Run 6: TODO

[x] Compilation completed successfully in 1 s 737 ms (moments ago)

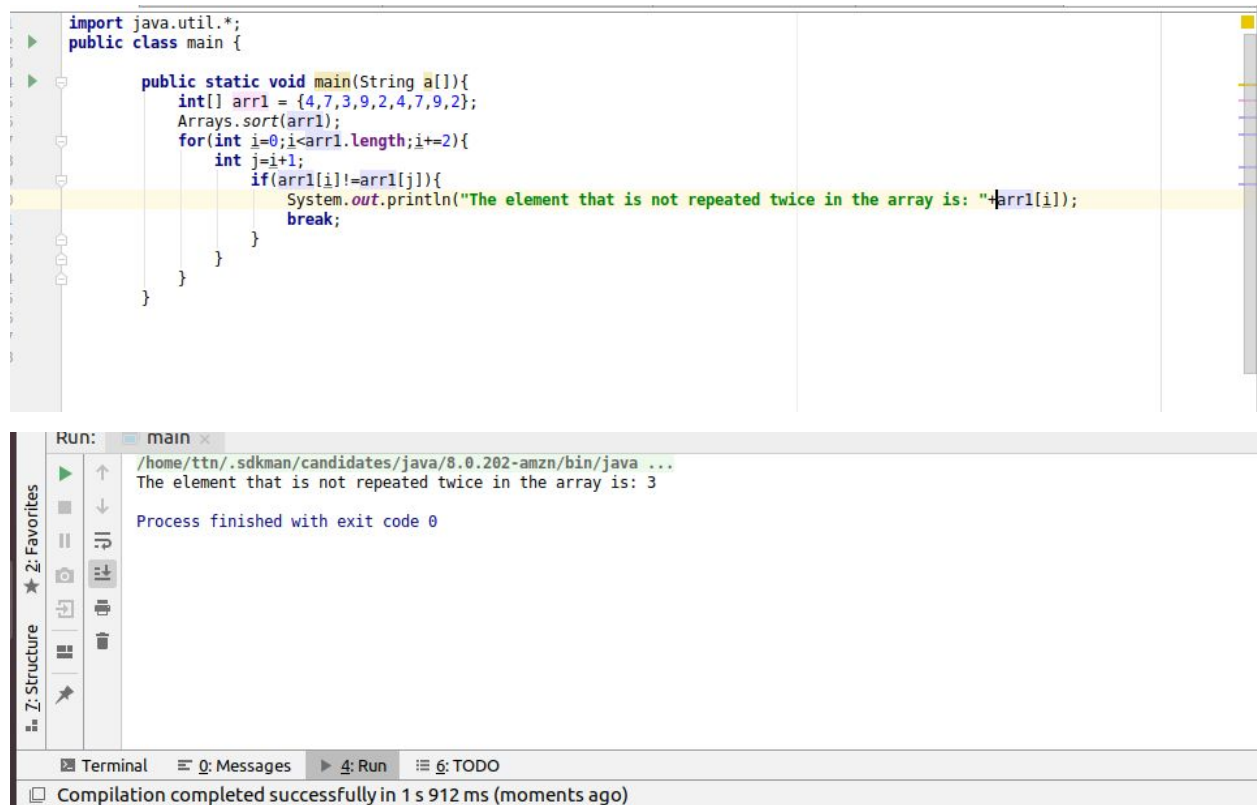
Q5. Find common elements between two arrays.

The screenshot displays an IDE with two main panels. The top panel shows a Java source file named 'main' with the following code:

```
1 import java.util.*;
2 public class main {
3
4     public static void main(String a[]) {
5         int[] arr1 = {4,7,3,9,2};
6         int[] arr2 = {3,2,12,9,40,32,4};
7         for(int i=0; i<arr1.length; i++){
8             for(int j=0; j<arr2.length; j++){
9                 if(arr1[i]==arr2[j]){
10                     System.out.println(arr1[i]);
11                 }
12             }
13         }
14     }
15 }
16
17
```

The bottom panel shows the terminal output of the program. The command executed is `/home/ttn/.sdkman/candidates/java/8.0.202-amzn/bin/java ...`. The output consists of the numbers 4, 3, 9, and 2, each on a new line. Below the output, it states "Process finished with exit code 0". The IDE's status bar at the bottom indicates "All files are up-to-date (moments ago)".

Q6. There is an array with every element repeated twice except one. Find that element



The image shows a screenshot of an IDE with two main panels. The top panel displays a Java program. The bottom panel shows the execution output.

```
import java.util.*;
public class main {

    public static void main(String a[]){
        int[] arr1 = {4,7,3,9,2,4,7,9,2};
        Arrays.sort(arr1);
        for(int i=0;i<arr1.length;i+=2){
            int j=i+1;
            if(arr1[i]!=arr1[j]){
                System.out.println("The element that is not repeated twice in the array is: "+arr1[i]);
                break;
            }
        }
    }
}
```

The bottom panel shows the execution output:

```
Run: main x
/home/ttn/.sdkman/candidates/java/8.0.202-amzn/bin/java ...
The element that is not repeated twice in the array is: 3
Process finished with exit code 0
```

The bottom panel also includes a status bar with the following information:

- Terminal
- 0: Messages
- 4: Run
- 6: TODO
- Compilation completed successfully in 1 s 912 ms (moments ago)

Q7. Write a program to print your Firstname,LastName & age using static block,static method & static variable respectively

```
main.java × FirstName.java × CommonElements.java × CountChar.java × Percent
1 public class FirstName {
2     public static String firstname;
3     static{
4         firstname="Japneet";
5     }
6     static String lastname="Kaur";
7
8     static int age=21;
9     static void disp(){
10        System.out.println("Age is: "+age);
11    }
12 }
13
14
```

```
main.java × FirstName.java × CommonElements.java × CountChar.java × Percent
1 import java.util.*;
2 public class main {
3
4     public static void main(String args[]) {
5         System.out.println("First Name: " + FirstName.firstname);
6         System.out.println("Last Name: " + FirstName.lastname);
7         FirstName.disp();
8     }
9 }
```

```
Run: main x
/home/ttn/.sdkman/candidates/java/8.0.202-amzn/bin/java ...
First Name: Japneet
Last Name: Kaur
Age is: 21

Process finished with exit code 0

Terminal 0: Messages 4: Run 6: TODO
Compilation completed successfully in 1 s 735 ms (moments ago)
```

Q8. Write a program to reverse a string and remove character from index 4 to index 9 from the reversed string using String Buffer

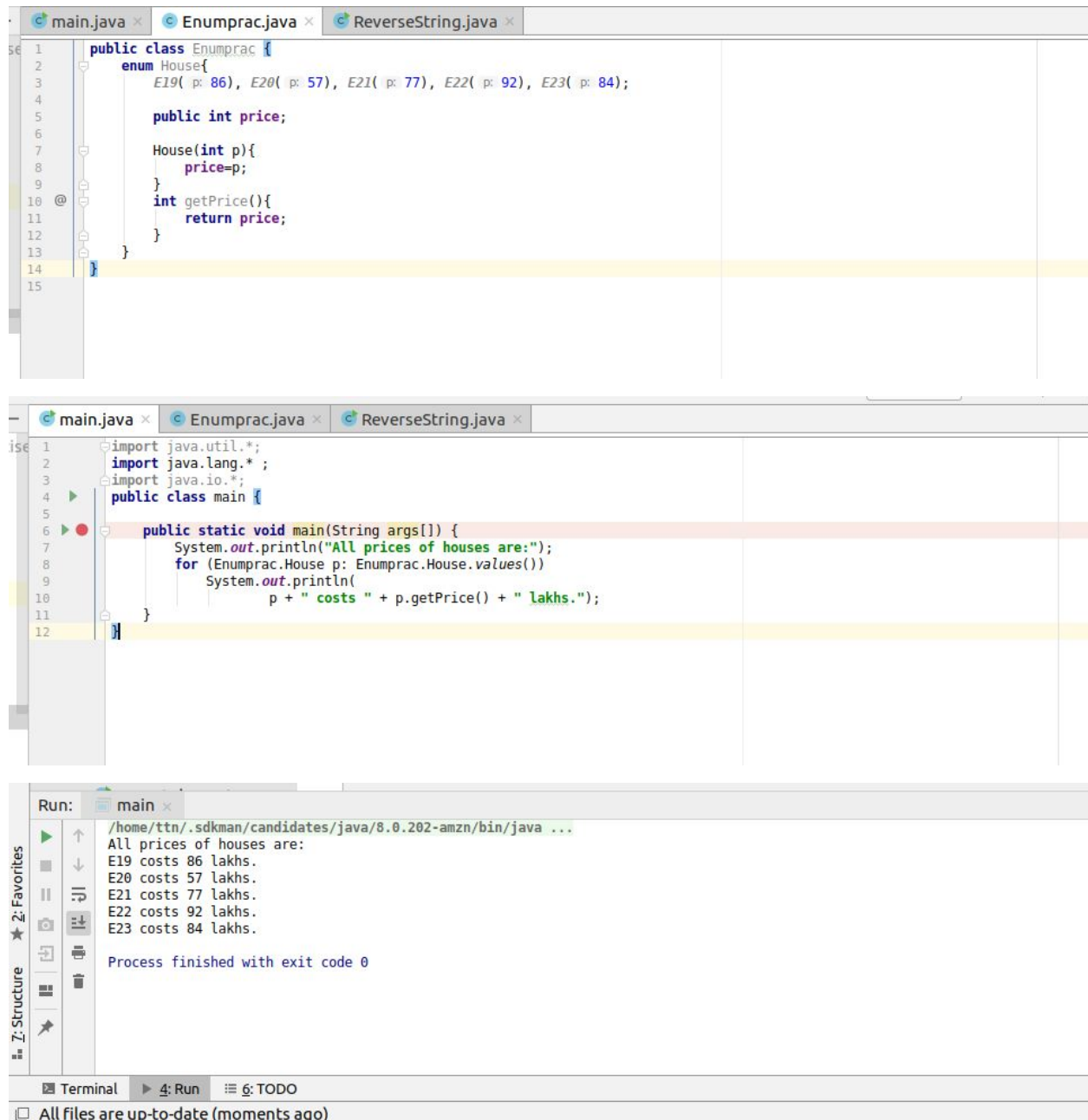
```
main.java ReverseString.java CountChar.java FirstName.java CommonElements.java
1 import java.util.*;
2 import java.lang.*;
3 import java.io.*;
4 public class main {
5
6     public static void main(String[] args) {
7         StringBuffer input= new StringBuffer("To The New Noida");
8         input=input.reverse();
9         System.out.println("The reversed String is: "+input);
10        input=input.replace( start: 4, end: 9, str: "");
11        System.out.println("The String after removing characters from index 4 to 9 is: "+input);
12    }
13 }
14
15
```

```
Run: main x
/home/ttn/.sdkman/candidates/java/8.0.202-amzn/bin/java ...
The reversed String is: adioN weN ehT oT
The String after removing characters from index 4 to 9 is: adio ehT oT

Process finished with exit code 0

Terminal 0: Messages 4: Run 6: TODO
Compilation completed successfully in 1 s 614 ms (moments ago)
```


Q9. Write a program to display values of enums using a constructor & getPrice() method (Example display house & their prices)



```
1 public class Enumpract {
2     enum House{
3         E19( p: 86), E20( p: 57), E21( p: 77), E22( p: 92), E23( p: 84);
4
5         public int price;
6
7         House(int p){
8             price=p;
9         }
10        int getPrice(){
11            return price;
12        }
13    }
14 }
15
```

```
1 import java.util.*;
2 import java.lang.*;
3 import java.io.*;
4 public class main {
5
6     public static void main(String args[]) {
7         System.out.println("All prices of houses are:");
8         for (Enumpract.House p: Enumpract.House.values())
9             System.out.println(
10                 p + " costs " + p.getPrice() + " Lakhs.");
11     }
12 }
```

Run: main x

```
/home/ttn/.sdkman/candidates/java/8.0.202-amzn/bin/java ...
All prices of houses are:
E19 costs 86 lakhs.
E20 costs 57 lakhs.
E21 costs 77 lakhs.
E22 costs 92 lakhs.
E23 costs 84 lakhs.

Process finished with exit code 0
```

Terminal | 4: Run | 6: TODO

All files are up-to-date (moments ago)

Q10. Write a single program for following operation using overloading

A) Adding 2 integer number

B) Adding 2 double

```
1 public class Sum {  
2  
3     @  
4     public static int sum(int x, int y){  
5         return(x+y);  
6     }  
7     @  
8     public static double sum(double x, double y){  
9         return(x+y);  
10    }  
11    public static void main(String[] args) {  
12        System.out.println(sum( x: 2, y: 4));  
13        System.out.println(sum( x: 2.3, y: 4.5));  
14    }  
15 }
```

Run: Sum x

```
/home/ttn/.sdkman/candidates/java/8.0.202-amzn/bin/java ...  
6  
6.8  
Process finished with exit code 0
```

Terminal 4: Run 5: Debug 6: TODO

All files are up-to-date (moments ago)

C) multiplying 2 float

D) multiplying 2 int

The screenshot shows an IDE with two tabs: 'Sum.java' and 'Multiply.java'. The 'Multiply.java' tab is active, displaying the following code:

```
1 public class Multiply {
2     @ public static int multiply(int x, int y){
3         return(x+y);
4     }
5     @ public static float multiply(float x, float y){
6         return(x+y);
7     }
8
9     public static void main(String[] args) {
10         System.out.println(multiply( x: 2, y: 4));
11         System.out.println(multiply( x: 2.3f, y: 4.3f));
12     }
13 }
14
15
16
```

Below the code editor, the 'Run' tab is active, showing the execution output:

```
Run: Multiply x
/home/ttn/.sdkman/candidates/java/8.0.202-amzn/bin/java ...
6
6.6000004
Process finished with exit code 0
```

The IDE interface includes a sidebar on the left with 'Favorites' and 'Structure' views, and a bottom status bar indicating 'All files are up-to-date (moments ago)'.

E) concate 2 string

```
1 public class Concat {
2     public static String s1="String1";
3     public static String s2="String2";
4
5     @ public static String concat()
6     {
7         return s1 + s2;
8     }
9
10    public static void main(String[] args) {
11        System.out.println("The concatenated strings are: " + s1 + s2);
12        System.out.println(concat());
13    }
14
15 }
16
17 }
```



```
Run: Concat x
/home/ttn/.sdkman/candidates/java/8.0.202-amzn/bin/java ...
The concatenated strings are:
String1String2

Process finished with exit code 0

Terminal 0: Messages 4: Run 5: Debug 6: TODO
Compilation completed successfully in 1 s 765 ms (moments ago)
```

F) Concat 3 String

```
1 public class Concat {
2     public static String s1="String1";
3     public static String s2="String2";
4     public static String s3="String3";
5
6     public static String concat()
7     {
8         return s1 + s2 + s3;
9     }
10
11     public static void main(String[] args) {
12         System.out.println("The concatenated strings are: ");
13         System.out.println(concat());
14     }
15
16 }
17
18
```

Run: Concat x

/home/ttn/.sdkman/candidates/java/8.0.202-amzn/bin/java ...
The concatenated strings are:
String1String2String3

Process finished with exit code 0

Terminal 0: Messages 4: Run 5: Debug 6: TODO

Compilation completed successfully in 1 s 769 ms (moments ago)

Q11.Create 3 sub class of bank SBI,BOI,ICICI all 4 should have method called getDetails which provide there specific details like rateofinterest etc,print details of every banks

```

1 public class Bank {
2     public void getDetails(){
3         System.out.println("The details are: ");
4     }
5 }
6 class SBI extends Bank{
7     int rateOfInterest=4;
8     int numberOfloans=2500;
9     int numberOfbranches=110;
10    public void getDetails(){
11        System.out.println("Rate of interest = " + rateOfInterest);
12        System.out.println("Number of loans given= " + numberOfloans);
13        System.out.println("Number of branches: "+numberOfbranches);
14    }
15 }
16 class BOI extends Bank {
17     int rateOfInterest = 3;
18     int numberOfloans = 1798;
19     int numberOfbranches = 97;
20 }
21 public void getDetails() {
22     System.out.println("Rate of interest = " + rateOfInterest);
23     System.out.println("Number of loans given= " + numberOfloans);
24     System.out.println("Number of branches: " + numberOfbranches);
25 }
26 }
27 }
28 }
29 class ICICI extends Bank {
30     int rateOfInterest = 4;
31     int numberOfloans = 2189;
32     int numberOfbranches = 256;
33 }
34 public void getDetails() {
35     System.out.println("Rate of interest = " + rateOfInterest);
36     System.out.println("Number of loans given= " + numberOfloans);
37     System.out.println("Number of branches: " + numberOfbranches);
38 }
39 }

```

```

1 public void getDetails() {
2     System.out.println("Rate of interest = " + rateOfInterest);
3     System.out.println("Number of loans given= " + numberOfloans);
4     System.out.println("Number of branches: " + numberOfbranches);
5 }
6 }
7 }
8 }
9 }
10 }
11 }
12 }
13 }
14 }
15 }
16 }
17 }
18 }
19 }
20 }
21 }
22 }
23 }
24 }
25 }
26 }
27 }
28 }
29 }
30 }
31 }
32 }
33 }
34 }
35 }
36 }
37 }
38 }
39 }
40 }
41 }
42 }
43 }
44 }
45 }
46 }
47 }
48 }
49 }
50 }
51 }
52 }
53 }
54 }
55 }
56 }
57 }
58 }
59 }
60 }
61 }
62 }
63 }
64 }
65 }
66 }
67 }
68 }
69 }
70 }
71 }
72 }
73 }
74 }
75 }
76 }
77 }
78 }
79 }
80 }
81 }
82 }
83 }
84 }
85 }
86 }
87 }
88 }
89 }
90 }
91 }
92 }
93 }
94 }
95 }
96 }
97 }
98 }
99 }
100 }

```

Run: ICICI x

2: Favorites

2: Structure

Run

Debug

TODO

Rate of interest = 4

Number of loans given= 2500

Number of branches: 110

Rate of interest = 3

Number of loans given= 1798

Number of branches: 97

Rate of interest = 4

Number of loans given= 2189

Number of branches: 256

Process finished with exit code 0

Terminal

4: Run

5: Debug

6: TODO

All files are up-to-date (a minute ago)