Uttara InfoSolutions

www.uttarainfo.com

Uttara Practicals - 5

Note: Go through the working of TestWrapper.java, TestArray. java & AddArrayElements.java first.

1) Create a Person class. A Person has a name (String) and age (int). Create a TestPerson class with main(). Create an object of person and see if you can access name and age directly (using reference) to set the value and print the values. Now mark the name and age in Person class as private. Now try to recompile TestPerson. Will compiler allow to access private marked variables to class user? Now create setter/getter methods to both name and age. Validate that name should not be null, blank and length of name should not be more than 30. Validate age such that it shouldn't be negative or > 100. Now in main() of tester class, invoke setter and try to pass bad values, compile and test the output. Are you getting correct error messages getting printed out? Now create a Person no-arg constructor and put an SOP in it in Person class:

```
public Person()
{
        SOP(“in Person no-arg constr”);
}
```

Recompile Person.java and run TestPerson. Do you see the SOP getting printed out? Recall when constructor body will be executed in object creation. Create another object of Person. Is the SOP printed again? How many times is a constructor executed per object? Add another 2 constructors in Person which accepts name as parameter and name,int as param:

```
public Person(String name)
{
        SOP(“in person string param constr”);
 // how to assign local variable name to instance variable name here?
}

public Person(String name, int age)
{
        SOP(“in person string,int param constr”);
        //assign the age,name local variable to instance variables.
}
```

Now create a Person object in main() by giving name, age as “Ramu”,10 to constructor. Comment out other code, compile and run. How many constructors got executed? What SOPs are you seeing and why?
Now I want you to validate name and age in constructors too. But think how to reuse the validation in both constructors and setters?

Next we need to keep a track of number of objects created of Person no matter which constructor is used to create the object. What kind of variable will you create count as (single copy?)? Where will you increment it so that no matter which constructor is executed, count gets incremented by one. Test by creating Person objects in a for loop of 100 and see if the count gets incremented as well.

2) Create a class Song. A song has a name (String) and lyrics (String). A song can be played. When you play, it prints out its lyrics to monitor. Create the class with 2 instance variables with 2 setter / getter methods (setName(String str), setLyrics(String n), getName(), getLyrics(). Make sure validate for null and empty string). Now create a TestSong class with main(). Create 2 objects of Song and set different names and lyrics. Invoke play() and verify if the songs are playing correctly. Now invoke setter methods to change the lyrics and invoke play() again. Has the lyrics changed or not?
Now create a paramterized constructor to accept both name and lyrics in Song class. Put an SOP in the constructor as well. Compile Song.java. Try to recompile TestSong.java. Is the compilation succeeding? Why not? Change your code in TestSong to create 2 song objects by passing parameters to constructors. Comment out the setter methods used. Now recompile and run the program. Do you understand the usage of Constructors now? Add another constructor to Song to be accept only name as parameter. Can we have multiple constructor in the same class? Put SOP in this constructor too. In TestSong, create another Song object by passing only name. Check by calling play() as to what is the lyrics being printed. Why so? Now invoke setLyrics() and pass a lyrics string as param. Invoke play() again on the same object. Do you now understand why to have setter methods still in a class that contains constructors? Make sure you validate in constructors too.

3) Create a static int variable val in a class named T and initialise it to 10. Code a main() and then print the value of val. Create a reference of T and point it to an object and then print T.val. Set the reference of T and point it to null and then print T.val. Do you see the output? Can you explain why? Create a method named test() as an instance method. Access val and print it there. Invoke test() from main(). Do you see val getting printed? Now create an instance int variable named p and initialise it to 40. Now in a static method, access p directly. Try to compile. Does it? Why not? How to access an instance variable in a static method?? In main(), directly print p. Can you?

4) Create a Joke class. A Joke has a text and has print functionality (text instance variable and print()). When you invoke print() on a Joke object, the text will be printed to the monitor. Mark the text variable private, expose setter/getter and have a parameterised constructor to pass in the text during Joke object creation time. Keep a track of Joke objects that are created. How will you create a single copy count variable? Where will you increment it? Test in a tester class by creating a few Joke objects and printing Joke.count to monitor.

5) *Important* A person has a name, Car, Dog and a favourite Song (reuse classes Car, Dog and Song from earlier problems given in last Lab). When you ask a Person to commute and give him a destination (String parameter), then he will start the car, drive the car and stop the car and print that he has reached the destination. When you ask the person to sing, he will sing (print) his favourite song with lyrics. When you ask a person to take a walk, he will take his dog for a walk and the dog will bark. Person has the ability to eat Food. Food has name and price. Food must be given to Person when you invoke eat(). When a person is asked to eat, he will specify that he is eating food with name and say out his name as well. Person also has a generatePrime() behaviour. When you give him a number as input, then he will generate all prime numbers until that number and print to monitor.

6) Test wrapper class usage => do you know how to convert a primitive into an object & back?, do you know how to access constants in Wrapper classes? Test the working of parsing & Character static methods.

7) Create a Math class with the following methods (overload methods - same method name with different argument lists just like constructor overloading):

i) Ability to add 2 ints
ii) Ability to add 2 doubles
iii) Ability to add 2 complex numbers. Create a Complex class with 2 int instance variables a and b. a represents the real part and b the imaginary.

8) Test whether an int number has an int square root or not (Use Math class) Ex: 16 square root is 4 which is an integer, 20 square root is 4.X which is not an int square root.

9) WAP to generate 10 random strings with 6 length each with each containing random digits.

10) Create the following static overloaded methods with simple SOPs in a class named X:
test(long)
test(double)
test(short)

Now try invoking the test() from main() by passing 5, 5.5, 5.5f,10L, a byte as parameter and understand which method body is being picked. Do you understand how the decision is made to pick which method body?

11) WAP to create an array with 10 size and of int holding ability. Store values 1-10 in it. Using the length variable, access each element of the array and print it to the console. Create the array in 2 ways - one using the new operator and then storing the values individually, two by creating the array with the values directly in the array. Loop over the array using an index variable.Also loop over it using for-each loop.
Ex:

```
int[] arr1 = new int[10]; // to create an empty array
int[] arr2 = {10,20,30}; // to create a literal array

for(int i = 0 ; i < arr1.length ; i++) // to loop over the array
        System.out.println("value in "+i+"th box = "+arr1[i]);

for(int val : arr2)
```

```
        System.out.println(val);
```

12) Basic String programs(all main() based programs):
a) Create a String object using new operator and using a string literal and print out its value using SOP to the console
b) Create a String and print out its length to the console using SOP
c) Take the input of a string from the command line (arg[0]) and print "You have a good name, <name concatenated>!" if the length of the string is < 12 chars and "You have a very long name, <name concatenated>!" if the length of the name is >=12 chars. If no input has been given, show an appropriate error message.
d) Take a string input from arg[0]. Print the chars one at a time to the console.
e) Take a string input from arg[0]. Print the even chars only.
f) Take a string input from arg[0]. Print the alternate chars in one line and the remaining in another.
g) Take string inputs from arg[0], arg[1]. Print "equal" if they have the same contents or "not equal" if they are not.
h) Take string inputs from arg[0], arg[1]. Print whichever string is bigger than the other alphabetically
i) Take string inputs from arg[0], arg[1]. Find out if arg[1] is present in arg[0] string and print if the search succeeds.
j) Take string inputs from arg[0], arg[1]. Convert their cases and print them to the console.

13)Code an add() method that will add all the numbers given and return the result (use array of ints as param to the add() method). See AddArrayElements.java if you have a doubt.

14) WAP (directly in main()) to create an array of strings with 5 colors as values. Use Math.random() to randomly print out 5 values from the array.

15) Person has a name and age. A person has a number of pet names(20max) which he obtains over a period of time. He can dance; if his age is less than 25 he can do chacha. If his age is greater than 25, he does the waltz. He can sing too and when he is asked to sing, he uses his petnames to form the song (randomly). Write a tester program to test persons.

See this below code only after trying to solve the problem on your first. This is important.

```
Person
        private String name;
        private int age;
        private String[] petNames = new String[20]; // since he can have max 20 names
        int count=0;

        public void sing()
        {
                String song = "";
                for(int i = 0; i < petNames.length; i++)
                {
                        int n = (int)(20 * Math.random());
                        song = song + petNames[n];
        }
        SOP(song);
    }
    public void addPetName(String n)
    {
        if(count < petNames.length)
            petNames[count++] = n;
        else
            SOP(..);
    }
    public boolean searchPetName(String n)
    {
        // search in the petNames array whether a name equal to n exists...and if
yes, return true, else return false;

        for(String s : petNames)
        {
            if(s.equals(n))
                return true;
        }
        return false;

    }
```

Understand the correct working of this solution properly. Take assistance of Lab Instructors if required. Ask specific questions.

16) WAM to pass 2 arrays of ints to a method. The method should return the max value present across both the array elements.

17) WAM to pass 2 arrays of ints to a method. The method should return the average of the values across the 2 arrays. The avg returned should be exact and not an approximation.

18)

In main()->

```
String str = “blah”;
str.toUpperCase();
SOP(str); what is printed and why?

str = “   blah   “;
SOP(str.trim()); //after this, print its length.

String s1 = new String(“abc”); // how many objects are getting created and where?
String s2 = new String(“abc”); // how many objects are getting created and where?
SOP(s1==s2); // do you understand the output?
SOP(s1.equals(s2)); // do you understand the output?
String s3 = “abc”;
```

```
SOP(s1==s3); // do you understand the output?
SOP(s1.equals(s3)); // do you understand the output?
String s4 = “abc”;
SOP(s3==s4); // do you understand the output?
SOP(s3.equals(s4)); // do you understand the output?
```

19) Create a method to accept a string as parameter and return how many vowels are present in it.

20) Create a method to accept a string sentence and reverse all its words in place and return it. Try different ways of reversing as discussed in class.

21) *important repeat problem* Create a method to accept 2 strings and find number of occurrences of second string in first string and return the result.

22) Create a method to accept a string as parameter and return a new string with all the unicode int values with a dot delimiter. Ex: input is abd, output should be 97.98.100

Methods to use:

Methods required:

```
1) int len = str.length()
2) for (int i = 0 ; i < str.length() ; i++)
        char c = str.charAt(i);
3) char[] arr = str.toCharArray();
4) String s1 = str.toUpperCase(); // str.toLowerCase();
5) boolean result = s1.equals(s2)
6) int pos = s1.indexOf(s2)
7) boolean result = s1.contains(s2)
8) int pos = s1.indexOf(<int pos>, s2) // -1 if search fails
9) String[] sa = str.split(" "); // splits the string on space token
10) int[] arr = {10,20,30}; // int literal array
11) String[] arr = new String[]{"rosey","posey”}; // String literal array
12) for(String s : arr)
        SOP(s);
13) for(int i = 0 ; i < arr.length ; i++)
        SOP(arr[i]);
```

Things to remember:

1) String is a class in Java (capital S)
2) length is a variable in an array and method in a string (arr.length , str.length())
3) You have to capture the returned value from a method to use it: String name = per.getName(); just calling per.getName() will not give you the value.
4) method names of String are fixed. You have to use proper naming convention (camel case).