**Program 1:Exercise 1:** Create a class with a method which can calculate the sum of first n natural numbers which are divisible by 3 or 5.

|  |  |
| --- | --- |
| Method Name | calculateSum |
| Method Description | Calculate Sum |
| Argument | int n |
| Return Type | int-sum |
| Logic | Calculate the sum of first n natural numbers which are divisible by 3 or 5. |

**Program2:Exercise 2:** Create a class with a method to find the difference between the sum of the squares and the square of the sum of the first n natural numbers.

|  |  |
| --- | --- |
| Method Name | calculateDifference |
| Method Description | Calculate the difference |
| Argument | int n |
| Return Type | int - Sum |
| Logic | Find the difference between the sum of the squares of the first n natural numbers and the square of their sum.  For Example if n is 10,you have to find  (1^2+2^2+3^2+….9^2+10^2)-  (1+2+3+4+5…+9+10)^2 |

**Program3:Exercise 3:** Create a class containing a method to create the mirror image of a String. The method should return the two Strings separated with a pipe(|) symbol .

|  |  |
| --- | --- |
| Method Name | getImage |
| Method Description | Generate the mirror image of a String and add it to the existing string. |
| Argument | String |
| Return Type | String |
| Logic | Accepts One String  Find the mirror image of the String  Add the two Strings together separated by a pipe(|) symbol.  For Example  Input : EARTH  Output : EARTH|HTRAE  Hint: Use StringBuffer API (Ex: For this problem reverse method in Stringbuffer can be used)  Note: Learn the other APIs in StringBuffer |

**Program4:Exercise 4:** Create a method to check if a number is an increasing number

|  |  |
| --- | --- |
| Method Name | checkNumber |
| Method Description | Check if a number is an increasing number |
| Argument | int number |
| Return Type | boolean |
| Logic | A number is said to be an increasing number if no digit is exceeded by the digit to its left.  For Example : 134468 is an increasing number |

**Program5:Example 5:** Create a method to check if a number is a power of two or not

|  |  |
| --- | --- |
| Method Name | checkNumber |
| Method Description | Checks if the entered number is a power of two or not |
| Argument | int n |
| Return Type | boolean |
| Logic | Check if the input is a power of two.  Ex: 8 is a power of 2 |

**Program6:Example 6:** A school offers medals to the students of tenth based on the following criteria

If(Marks>=90) : Gold

If(Marks between 80 and 90) : Silver

If(Marks between 70 and 80) : Bronze

Note: Marks between 80 and 90 means marks>=80 and marks<90

Write a function which accepts the marks of students as a Hashmap and return the details of the students eligible for the medals along with type of medal.

The input hashmap contains the student registration number as key and mark as value.

The output hashmap should contain the student registration number as key and the medal type as value.

|  |  |
| --- | --- |
| Method Name | getStudents |
| Method Description | Generate the list of students eligible for scholarship |
| Argument | Hashmap |
| Return Type | Hashmap |
| Logic | The method should return the details of the students eligible for the medals along with the medal type. |

**Program9:**

**Example 9:** Create a method which can perform the following operations on two String objects S1 and S2. The output of each operation should be added to an arraylist and the arraylist should be returned.(Assume S2 is of smaller size)

Examples for below statements are shown in the Logic part

1. Character in each alternate index of S1 should be replaced with S2

2. If S2 appears more than once in S1, replace the last occurrence of S2 in S1 with the reverse of S2, else return S1+S2

3. If S2 appears more than once in S1, delete the first occurrence of S2 in S1, else return S1

4. Divide S2 into two halves and add the first half to the beginning of the S1 and second half to the end of S1.

Note: If there are odd number of letters in S2, then add (n/2)+1 letters to the beginning and the remaining letters to the end. (n is the number of letters in S2)

5. If S1 contains characters that is in S2 change all such characters to \*

|  |  |
| --- | --- |
| Method Name | modifyStrings |
| Method Description | Perform the above mentioned actions on a String |
| Argument | String,String |
| Return Type | Arraylist |
| Logic | Do the above mentioned actions on the entered String.  For Example  S1=”JAVAJAVA”  S2=”VA’  1. **VA**A**VA**A**VA**A**VA**A (J replaced with VA, V replaced with VA etc.)  2. JAVAJAAV  3. JAJAVA  4. VJAVAJAVAA |

**Program10:**

**Example 10:** Create a method that accepts a number and modifies it such that the each of the digit in the newly formed number is equal to the difference between two consecutive digits in the original number. The digit in the units place can be left as it is.

Note: Take the absolute value of the difference. Ex: 6-8 = 2

|  |  |
| --- | --- |
| Method Name | modifyNumber |
| Method Description | Accepts a number and modify it as per the requirement |
| Argument | int number1 |
| Return Type | int |
| Logic | Accept a number and modify it such that the each of the digit in the newly formed number is equal to the difference between two consecutive digits in the original number.  For example.  Input: 45862  Output:13242  **Algorithm:**   Convert number into String   Extract each char using charAt method   Convert char to int and find the difference   Create new StringBuffer object and keep adding the difference   Finally convert StringBuffer to int |

**Program11:**

**Example 11:** Create a method which accepts the date of birth of person and date format and print the day (SUNDAY, MONDAY…) on which he was born.

Note: The output should be in upper case

|  |  |
| --- | --- |
| Method Name | getDayofWeek |
| Method Description | Finds the day of the week in which a person is born |
| Argument | String date, String dateFormat |
| Return Type | String – Day of week |
| Logic | Use Calendar API and switch case to get the day of the week  Ex: Input1 = 25/06/2012  Input2 = dd/MM/yyyy  Output= MONDAY |

**Program12:**

**Example 12:** You are asked to create an application for registering the details of jobseeker. The requirement is:

Username should always end with **\_job** and there should be atleast minimum of 8 characters to the left of **\_job**. Write a function to validate the same. Return true in case the validation is passed. In case of validation failure return false.

|  |  |
| --- | --- |
| Method Name | validateUserName |
| Method Description | Checks if the username is valid |
| Argument | String userName |
| Return Type | boolean |
| Logic | Checks if the username ends with \_job and contains at least 8 characters to the left of \_job. If valid return true. Else return false. |

**Program13:**

**Example 13:** Create a method that can accept an array of String objects and sort in alphabetical order. The elements in the left half should be completely in uppercase and the elements in the right half should be completely in lower case. Return the resulting array.

Note: If there are odd number of String objects, then (n/2)+1 elements should be in UPPPERCASE

|  |  |
| --- | --- |
| Method Name | getArrayList |
| Method Description | Converts the String array to ArrayList and sorts it |
| Argument | String []elements |
| Return Type | String [] modifiedArray |
| Logic | Load the elements in to an ArrayList ,sort it, convert the left half element to uppercase and right half elements to lower case .  Hint :  1. Use Collection  2. Use String API |

**Program14 and 15:**

**Example 14:** Create a method which can remove a List from another List

|  |  |
| --- | --- |
| Method Name | removeElements |
| Method Description | Removes the elements in one list that is present in the second list also. |
| Argument | List<String> list1, List<String> list2; |
| Return Type | List- ArrayList contains the resulting List after the removal process. |
| Logic | Accept two List objects list1 and list2 and remove the elements from list1 that are present in list2. This should be done in single step process without using loop.  Hint: Use the List API which removes all the items in List1 which are contained in List2 |

**Example 15:** Create a method which can remove all the elements from a list other than the list of elements specified.

|  |  |
| --- | --- |
| Method Name | removeElements |
| Method Description | Remove all the elements from a list other than the list of elements specified. |
| Argument | List<String> list1, List<String> list2; |
| Return Type | List- ArrayList contains the resulting List after the removal process. |
| Logic | Accept two List objects list1 and list2 and remove all the elements from list 1 other than the elements contained in list2.This should be done in single step process without using loop.  **Hint**: Use the List API method which can retain the elements available in the second list only |

**Program16:Example 16:** Create a method which accepts an array of numbers and returns the numbers and their squares in an HashMap

|  |  |
| --- | --- |
| Method Name | getSquares |
| Method Description | Accepts a list of numbers and return their squares |
| Argument | int[] |
| Return Type | Map |
| Logic | Iterate through the list, find the square of each number and add the elements to a map object with the number as the key and the square as the value. |

**Program17:Example 17:** Create a method which accepts the id and the age of people as a Map and decide if they are eligible for vote. A person is eligible for vote if his age is greater than 18. Add the IDs of all the eligible persons to list and return the list. (Assume date is in DD/MM/yyyy format)

|  |  |
| --- | --- |
| Method Name | votersList |
| Method Description | Generate the list of voters based on the ages of the people |
| Argument | Map |
| Return Type | List |
| Logic | Accept a map with ID as key and Date of Birth as value and check if the person is eligible to vote. A person is eligible for vote for if his age is greater than 18. If the person is eligible add his ID to the list.  Hint: Use Calendar API and SimpleDateFormat |

**Program18:Example 18:** Create a method which accepts an integer array, reverse the numbers in the array and returns the resulting array in sorted order

|  |  |
| --- | --- |
| Method Name | getSorted |
| Method Description | Return the resulting array after reversing the numbers and sorting it |
| Argument | int [] |
| Return Type | int |
| Logic | Accept and integer array, reverse the numbers in the array, sort it and return the resulting array. Hint :  1. Convert the numbers to String to reverse it  2. Use Collection APIs to sort it  **Ex:** {12,23,96,45}  **Step 1:** Reverse numbers {21,32,69,54}  **Step2:** Sort it {21,32,54,69}  **Hint**: Use String to reverse number  To sort it, Convert array to ArrayList and use Collections.sort |

**Program19:Example 19:** Create a method which accepts an integer array and removes all the duplicates in the array. Return the resulting array in descending order

|  |  |
| --- | --- |
| Method Name | modifyArray |
| Method Description | Remove duplicates |
| Argument | int [] |
| Return Type | int [] |
| Logic | Remove the duplicate elements in the array and sort it in descending order  Hint:  1. Use Collection API (TreeSet) to remove duplicates and sort the result in ascending order  2. Create a new array, iterate through elements in TreeSet and add it in the reverse order |

**Program:20Example 20:** Create a method that accepts a character array and count the number of times each character is present in the array. Add how many times each character is present to a hash map with the character as key and the repetitions count as value

|  |  |
| --- | --- |
| Method Name | countCharacter |
| Method Description | Count the number of occurrence of each character in a Character array |
| Argument | char[] |
| Return Type | map |
| Logic | Count the number of times each character appears in the array. Add the details into a hash map with character as key and count as value.  Example:  {‘A’,’P’,’P’,’L’,’E’}  Output: Will be hashmap with the following contents{‘A’:1,’P’:2,’L’:1,’E’:1} |

**Program21:Example 21:** A String contains a list of states and capitals. Write a method which can parse the string and return the states and capitals as map with state as key and capital as value.

The String is in the below format.

The state and capital is separated by a delimiter (del1). There will be multiple state-capital pairs and each state – capital pair is separated by another delimiter (del2).

Ex: Input will be **tamilnadu||chennai-karanataka||bengaluru.**

**Here, || will be provided as del1 and - will be provided as del2.**

|  |  |
| --- | --- |
| Method Name | getStates |
| Method Description | Accepts the states and capitals as a String and return a map |
| Argument | String data, char del1,char del l2 |
| Return Type | Map |
| Logic | Parse the string based on the delimiters and load it to a map with the state name as key and capital as value.  Hint: Use Stringtokenizer or split method in String class.  Try both the above ways to get familiarized with both APIs |

**Program22:Example 22:** In a certain television game show, a couple is considered as a perfect couple if both the husband’s and wife’s name contains the same set of characters. Each couple is provided with an ID. Write a method which can accept a Hashmap with ID as key and the husband’s and wife’s name separated with “-” as value. The method should generate the list of perfect couples based on the above mentioned criteria and return their IDs as List object.

|  |  |
| --- | --- |
| Method Name | checkPerfectCouple |
| Method Description | Select the set of perfect couples |
| Argument | Map |
| Return Type | List |
| Logic | Accept the Map  Iterate through it  Separate the husband’s and wife’s names  If they contain the same characters, add the ID to the List object.  Ex: Assuming VIMAL-MALIV is the value, this is a perfect couple since both these names contains same characters (in different order). |

**Program23:Example 23:** Create a method which can perform a particular String operation based on the user’s choice. The method should accept the String object and the user’s choice and return the output of the operation.

Options are

A: Add the String to itself

B: Replace alternate positions with \*

C: Remove duplicate characters in the String

D: Change alternate characters to upper case

|  |  |
| --- | --- |
| Method Name | changeString |
| Method Description | Modify the string based on user choice |
| Argument | String string, char ch |
| Return Type | String |
| Logic | Perform the required operation based on the user choice and return the resulting string |

**Program24:Example 24:** Create a method that accepts a String and checks if it is a positive string. A string is considered a positive string, if on moving from left to right **each** character in the String comes after the previous characters in the Alphabetical order.

For Example

ANT is a positive String (Since T comes after N and N comes after A)

APPLE is not positive since L comes before P in the alphabetical order.

The method should return true if the entered string is positive

|  |  |
| --- | --- |
| Method Name | checkPositive |
| Method Description | Checks if a String is positive |
| Argument | String |
| Return Type | boolean |
| Logic | Check if a string is positive based on the above criteria and return true if positive. Hint:  **Step 1:** Convert to Char array.  **Step 2:** Iterate through array, subtract 1st two characters (A-N). This will give the ASCII difference  **Step 3:** If result is negative, then return false and break. Else continue to next loop |

**Program:25: Example 25:** Create a method which accepts two Arraylist containing characters. Merge both arrays lists, sort the elements in the resulting list and return the resulting array.

|  |  |
| --- | --- |
| Method Name | mergeData |
| Method Description | Merge two arraylist , sort it and return the result as an integer array. |
| Argument | List, List |
| Return Type | char[] |
| Logic | Merge both arrays lists, sort the elements in the resulting list and return it as a char array. |