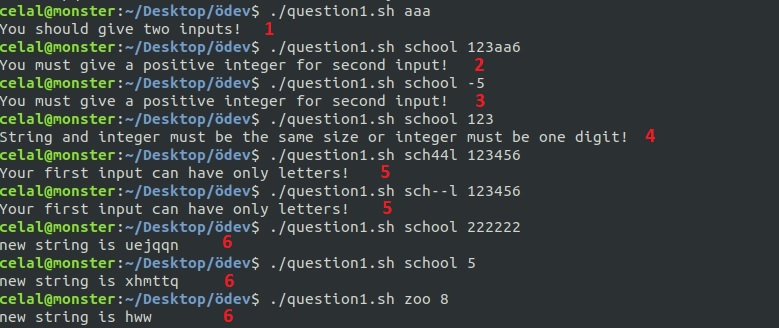
**SHELL SCRIPT PROGRAMS**

Program 1

In our first program, we take 2 inputs from the user; first one is a string and second one is an integer. If user doesn’t give 2 inputs, then program gives an error message (1). We take every index of that integer number and add them into an array. Then we check whether if there is a character other than a number in that array. If there is a letter in the second input, or the input is a negative number then program will give an error message (2-3).

We have some if-else statements that controls if the sizes of the inputs are equal or the second input is a one-digit number. If there is a case other than these two, it gives an error (4). In these if-else statements we also check that if the first input has only letters in it. Letters can be lower or upper case, if there is a character other than these it gives an error (5).

If there is no error, than the program prints the new string (6).



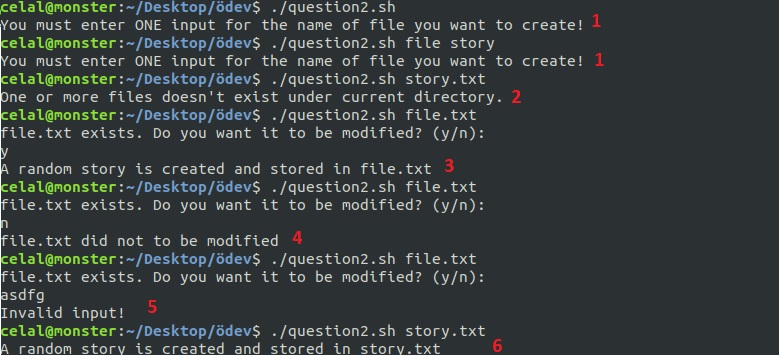
Program 2

In our second program, we take one input from the user for the name of the file he/she wants to create and store the random story in.

Firstly we check if he/she enter one input. If there is more than one input or there is none, then program gives an error message (1). If the input number is correct then we control that the text files we need are exists. If one or more files are missing then it gives an error message (2).

If all the files exist, then we read the texts and write the lines of them into three different arrays. Then we check if there is a file with the same name that the user gave us before. If there is, we ask user to modify it. If user says ‘yes’ we create a random story and write it into that file (3), if he/she says ‘no’ we give a message and don’t modify it (4). If the user’s answer is not one of these, then the program gives an error message (5).

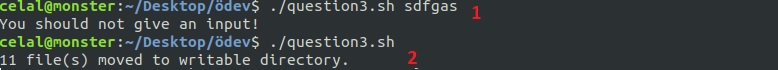
If there is no error and there isn’t a file with the same name, we create a file and store the story in it (6).



Program 3

In our third program, first we check if the user enters one input. If he/she does, then program will print an error (1).

If there is not a directory named ‘writable’ under the current directory then we create it, if there is a ‘writable’ directory then we just moved the files with write permission for user into it. We also count the number of files moved and print it (2).

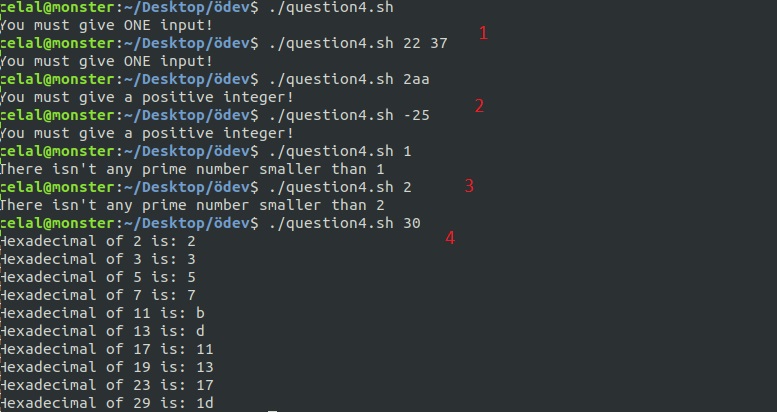


Program 4

In our fourth program, first we check if the user enters one input. If he/she doesn’t, then program will print an error (1).

We take every index of the input and add them into an array. Then we check whether if there is a character other than a number in that array. If there is a letter in the second input, or the input is a negative number then program will give an error message (2).

If user gives 1 or 2, then program will print a warning message (3).

If there is no error then we print the prime numbers smaller than the input and their hexadecimal values (4). 

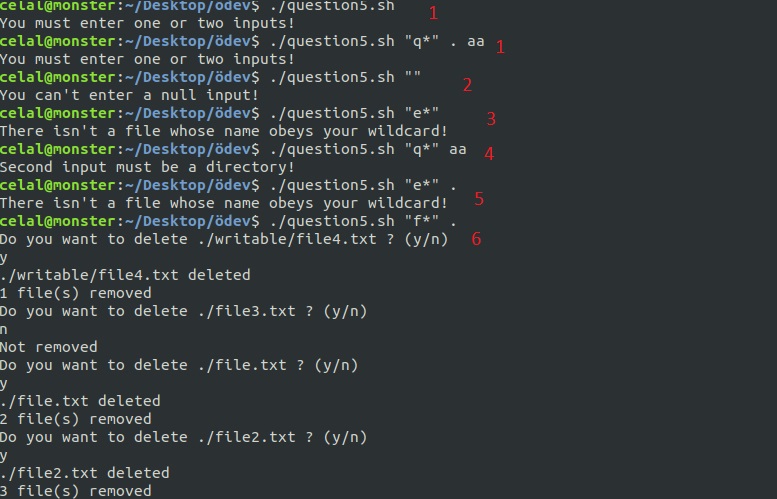
Program 5

In our fifth program, we take 1 or 2 inputs from the user. If there is more than two inputs or there is none, then program gives an error message (1).

If user gives one input, first we check if it is null or if it exists in the current directory. If it is null or doesn’t exist, then program gives an error message (2-3).

If user gives two inputs, we check if the second one is a directory. If it isn’t, then program gives an error (4). If it is, then we check if there is a file named as first input in the current directory and its subdirectories. If there isn’t, then program gives an error (5).

If the files exist, then we ask user to delete them (6).



Bonus(Menu)

We also write a menu named myprog.sh for our all programs. When user selects an option, we ask for the arguments we need.

If we execute the fifth program directly on the terminal we have to enter first input in double quotes; however if we call it by the menu we don’t use double quotes.

