1)

Write a program to print the following pattern

\* \* \* \* \* \* \* \* \*

\*

\*

\*

\*

\*

\*

\*

\*

\*

\*

\*

\*

\*

\* \* \* \* \* \* \* \* \*

import 'dart:io';

void main() {

print("Enter number of rows: ");

int row = int.parse(stdin.readLineSync()!);

print("Generated Hollow Hourglass Pattern is: ");

// Upper-half

for (int i = row; i > 0; i--)

{

for (int j = 0; j < row - i; j++) {

stdout.write(" ");

}

for (int j = 1; j < 2 \* i; j++)

{

if (i == 1 || i == row || j == 2 \* i - 1)

{

stdout.write("\*");

}

else

{

stdout.write(" ");

}

}

print("");

}

// // Lower-half

for (int i = 2; i <= row; i++) {

for (int j = 0; j < row - i; j++) {

stdout.write(" ");

}

for (int j = 1; j < 2 \* i; j++) {

if (i == 1 || i == row || j == 2 \* i - 1)

{

stdout.write("\*");

} else {

stdout.write(" ");

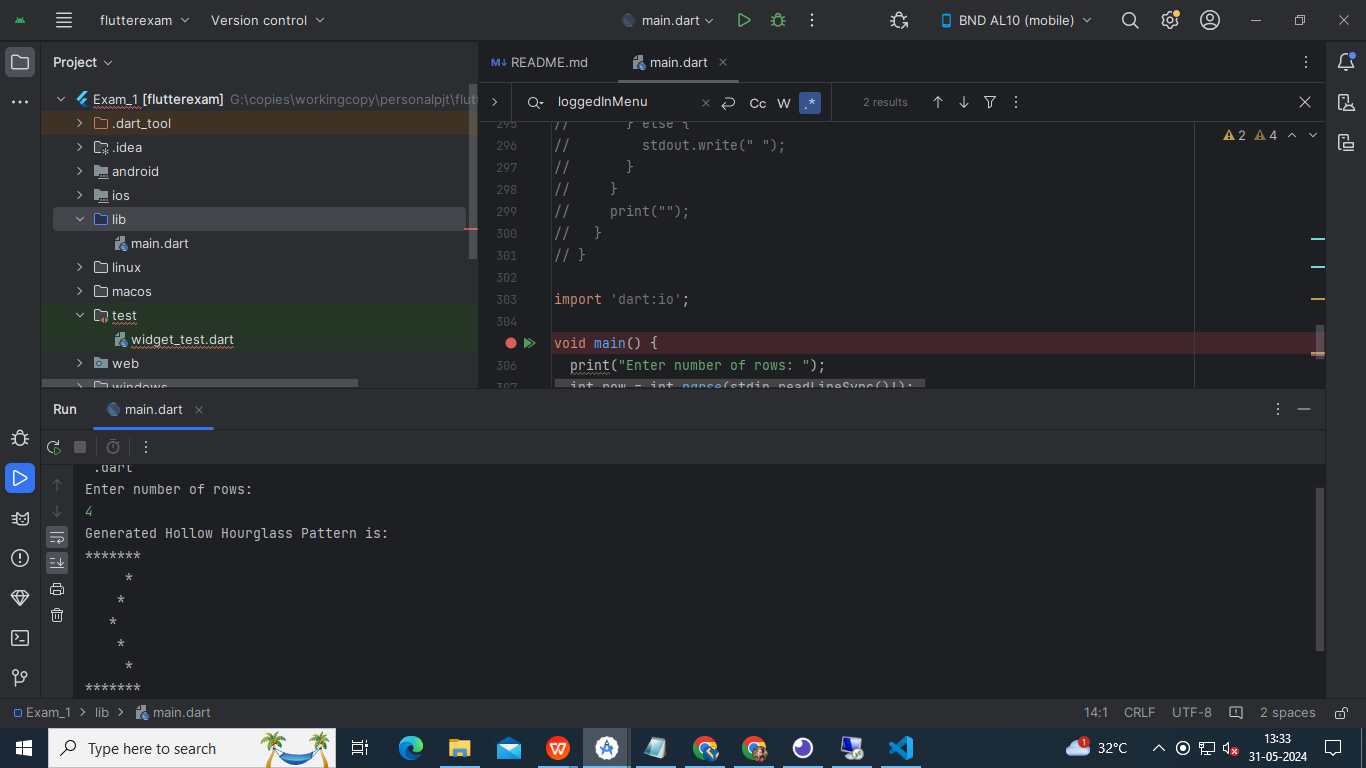
}

}

print("");

}

}



1. Write a Dart program that accepts integers into a list until 0 is entered. Generate separate lists for odd and even numbers from that list and Print the greatest number in the main list.

import 'dart:io';

void main() {

// Initialize empty lists

final numbers = <int>[];

final evenNumbers = <int>[];

final oddNumbers = <int>[];

int? ourInput;

// Loop to accept integers

do {

print('Enter an integer :');

ourInput = int.tryParse(stdin.readLineSync() ?? '');

if (ourInput != null) {

numbers.add(ourInput);

}

} while (ourInput != 0);

// Separate even and odd numbers

for (final number in numbers) {

if (number % 2 == 0) {

evenNumbers.add(number);

} else {

oddNumbers.add(number);

}

}

// Find the greatest number

final greatestNumber = numbers.reduce((a, b) => a > b ? a : b);

// Print results

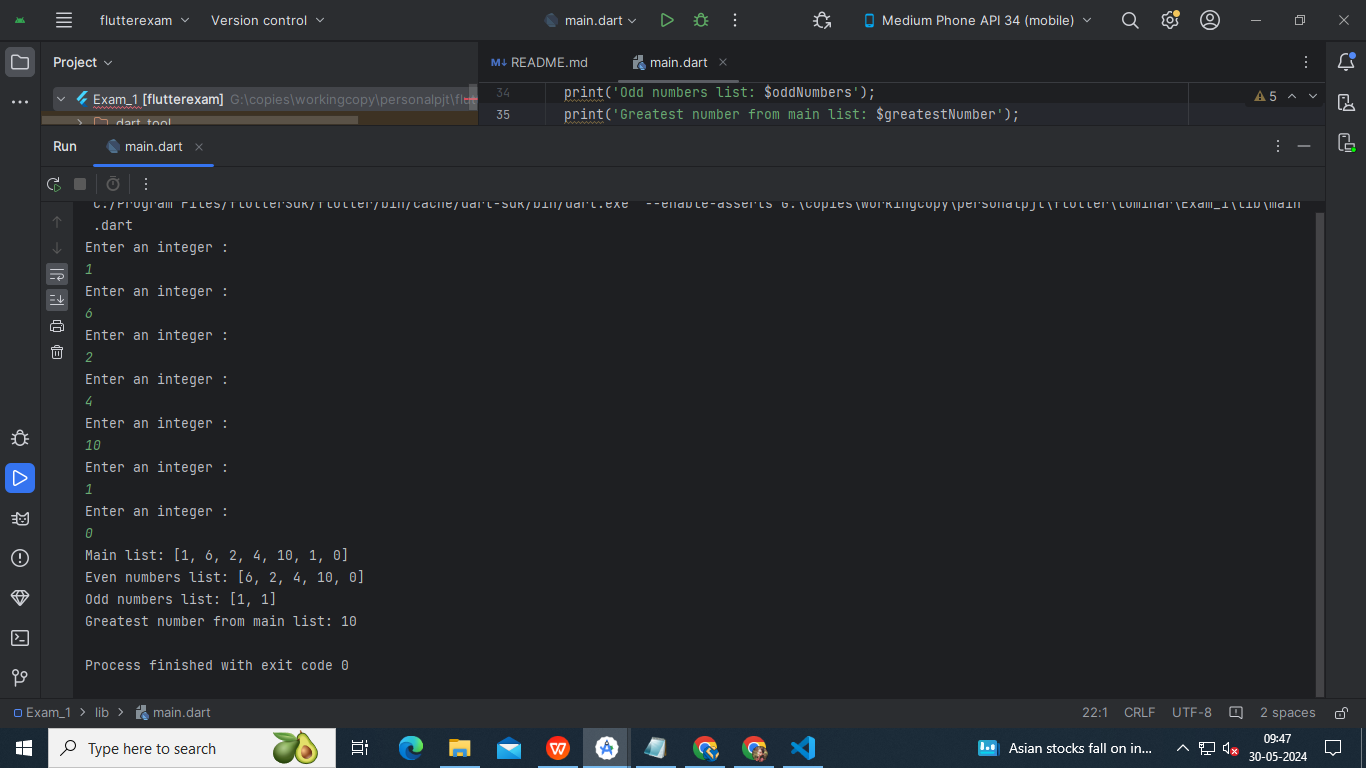
print('Main list: $numbers');

print('Even numbers list: $evenNumbers');

print('Odd numbers list: $oddNumbers');

print('Greatest number from main list: $greatestNumber');

}



1. Create a banking application which have user registration , login, bank details, deposit and withdrawal option and print the curresponding result with proper balance , use functions

void main() {

var users = <String, String>{};

String presentUser;

double balance = 1000.00;

int choice;

do {

print("Welcome to the Banking Application!");

print("1. Register");

print("2. Login");

print("3. Exit");

print("Enter your choice: ");

choice = int.parse(readLineSync(stdin)!);

switch (choice) {

case 1:

register(users);

break;

case 2:

presentUser = login(users)!;

if (presentUser != null) {

loggedInMenu(balance, presentUser);

}

break;

case 3:

print("Goodbye! Thank you for using the Banking Application!");

break;

default:

print("Invalid choice. Please try again.");

}

} while (choice != 3);

}

void register(Map<String, String> users) {

print("Enter your desired username: ");

String username = readLineSync(stdin)!;

print("Enter your password: ");

String password = readLineSync(stdin)!;

if (users.containsKey(username)) {

print("Username already exists. Please try again.");

} else {

users[username] = password;

print("Registration successful!");

}

}

String? login(Map<String, String> users) {

print("Enter your username: ");

String username = readLineSync(stdin)!;

print("Enter your password: ");

String password = readLineSync(stdin)!;

if (users.containsKey(username) && users[username] == password) {

print("Login successful!");

return username;

} else {

print("Invalid username or password. Please try again.");

return null;

}

}

void loggedInMenu(double balance, String presentUser) {

int choice;

do {

print("Welcome, $presentUser!");

print("1. Bank Details");

print("2. Deposit");

print("3. Withdraw");

print("4. Logout");

print("Enter your choice: ");

choice = int.parse(readLineSync(stdin)!);

switch (choice) {

case 1:

print("Account Balance: $balance");

break;

case 2:

print("Enter the amount to deposit: ");

double depositAmount = double.parse(readLineSync(stdin)!);

balance += depositAmount;

print("Deposit successful!");

print("Account Balance: $balance");

break;

case 3:

print("Enter the amount to withdraw: ");

double withdrawAmount = double.parse(readLineSync(stdin)!);

if (withdrawAmount > balance) {

print("Insufficient funds.");

} else {

balance -= withdrawAmount;

print("Withdrawal successful!");

print("Account Balance: $balance");

}

break;

case 4:

print("Logged out successfully!");

presentUser = "";

break;

default:

print("Invalid choice. Please try again.");

}

} while (choice != 4);

}

String readLineSync(Stdin stdin) => stdin.readLineSync()!;

