

Q1) Print the first N Fibonacci numbers (N is specified by the user)

A:

```
import 'dart:io';

void fibo(int n) {
  int n1=0,n2=1,n3;
  if (n>0){
    stdout.write(n1);
  }
  if (n>1){
    stdout.write(', $n2');
  }
  if (n>2){
    for(int i = 3 ;i <= n; i++){
      n3 = n1+n2;
      stdout.write(', $n3');
      n1 = n2;
      n2 = n3;
    }
  }
  if (n<=0){
    print('Invalid input!!');
  }
}

void main(){
  print('FIBONACCI GENERATOR\n');
  stdout.write('Enter number of terms:');
  int Num = int.parse(stdin.readLineSync());
  fibo(Num);
}
```

```
print('\n\nX-X-X-X-X-X-X-X-X-X-X-X-X-X-X-X');
}
```

Q x

Enter number of terms:15

X-X

Q x

Enter number of terms:2

X-X

Enter number of terms:0

X-X

Q2) Write a program to check whether the given number is semiprime or not

A:

```
import 'dart:io';
import 'dart:math';

bool Semiprime_check(int n) {
    int count=0;
    double sq=sqrt(n);
    for (int i=2 ; i<=sq+1 ; i++){
        while (n%i==0){
            n=n~/i;
            count+=1;
        }
        if (count>=2){
            break;
        }
    }
    if(n>1){
        count+=1;
    }

    if (count==2){
        return true;
    } else {
        return false;
    }
}

void main(){
```

```
> run-project SEMIPRIME CHECKER  
  
Enter number to be checked:10  
  
10 is semiprime  
  
X-X-X-X-X-X-X-X-X-X-X-X-X-X-X-X-X-X-X  
>
```

Q3) Check whether the sum of prime elements of the array is prime or not

A:

```
import 'dart:io';
import 'dart:convert';
import 'dart:math';

bool Prime_check(int n) {
  if(n<=1){
    return false;
  }

  for (int i = 2;i<=sqrt(n);i++){
    if(n%i == 0){
      return false;
    }
  }
  return true;
}

void main(){
  int Sum=0;
  print('PRIME SUM PROBLEM\n');
  stdout.write('Enter number of elements in array:');
  int Length = int.parse(stdin.readLineSync());
  var Array=new List();
  for (int a=1;a<=Length;a++) {
    stdout.write('\nEnter element $a:');
    int Element = int.parse(stdin.readLineSync());
    if (Prime_check(Element)) {
      Sum=Sum+Element;
    }
  }
}
```

```

    }
    Array.add(Element);
}

print("\nThe array has been created and traversed...");

print('The Array is $Array');

print('The sum of prime numbers=$Sum');

if (Prime_check(Sum)) {
    print('The sum is prime');
}

else {
    print('The sum is not prime');
}

print("\n\nX-X-X-X-X-X-X-X-X-X-X-X-X-X-X-X");
}

```

```

> run-project
PRIME SUM PROBLEM

Enter number of elements in array:5

Enter element 1:3

Enter element 2:6

Enter element 3:4

Enter element 4:11

Enter element 5:23

The array has been created and traversed...
The Array is [3, 6, 4, 11, 23]
The sum of prime numbers=37
The sum is prime

X-X-X-X-X-X-X-X-X-X-X-X-X-X-X-X
>

```

Q4) Create a courses module in dart with two classes BranchElective and OpenElective with common fields being courseName and courseCode. BranchElective has two extra fields which are branch and year. The following features are to be implemented

- a. An admin user can add courses(BranchElective, OpenElective) to the module
- b. A student can view branch electives based on his/her branch and year
- c. Any student can view open electives

A:

```
import 'dart:io';

class OpenElective{
    String courseCode;
    String courseName;
}

class BranchElective{
    String courseCode;
    String courseName;
    String branch;
    int year;
}

var OEL = List();
var BEL = List();

void Display(){
    stdout.write("\nEnter your branch:");
    String B = stdin.readLineSync();
    stdout.write("\nEnter your year:");
    int Y = int.parse(stdin.readLineSync());
    print("\nOPEN ELECTIVES");
    for (int i=0;i<OEL.length;i++) {
        print('${OEL[i].courseCode} -- ${OEL[i].courseName}');
    }
    print("\nBRANCH ELECTIVES for $B, year$Y");
}
```

```

for (int i=0;i<BEL.length;i++) {
    if (BEL[i].branch==B && BEL[i].year==Y){
        print('${BEL[i].courseCode} -- ${BEL[i].courseName}');
    }
}
}

```

```

void Create(){
    print('\nEnter course type: 1.Open Elective 2.Branch Elective');
    int ch = int.parse(stdin.readLineSync());
    if (ch==1){
        var Entry = new OpenElective();
        stdout.write('\nEnter course name:');
        Entry.courseName=stdin.readLineSync();
        stdout.write('\nEnter course code:');
        Entry.courseCode=stdin.readLineSync();
        OEL.add(Entry);
    }
    else if (ch==2){
        var Entry = new BranchElective();
        stdout.write('\nEnter course name:');
        Entry.courseName=stdin.readLineSync();
        stdout.write('\nEnter course code:');
        Entry.courseCode=stdin.readLineSync();
        stdout.write('\nEnter branch:');
        Entry.branch=stdin.readLineSync();
        stdout.write('\nEnter year:');
        Entry.year=int.parse(stdin.readLineSync());
        BEL.add(Entry);
    }
}

```



```
}  
}  
  
void main(){  
    int Active=1;  
    while(Active==1){  
        print("\nEnter type of user: 1.Admin 2.Student");  
        int UType=int.parse(stdin.readLineSync());  
        if (UType==1) {  
            Create();  
        }  
        else if (UType==2){  
            Display();  
        }  
        stdout.write("\nOperation complete\n1.Cont 2.Exit:");  
        Active=int.parse(stdin.readLineSync());  
    }  
    print('Exiting PGM...');  
}
```

run-project

Enter type of user: 1.Admin 2.Student
1

Enter course type: 1.Open Elective 2.Branch Elective
1

Enter course name:Creative Writing

Enter course code:ENG01

Operation complete

1.Cont 2.Exit:1

Enter type of user: 1.Admin 2.Student
1

Enter course type: 1.Open Elective 2.Branch Elective
1

Enter course name:Home Science

Enter course code:HSC01

Operation complete

1.Cont 2.Exit:1

Enter type of user: 1.Admin 2.Student
1

Enter course type: 1.Open Elective 2.Branch Elective
2

Enter course name:Python Programming

Enter course code:CS297

Enter branch:CS

Enter year:1

Operation complete

1.Cont 2.Exit:1

Enter type of user: 1.Admin 2.Student
1

Enter course type: 1.Open Elective 2.Branch Elective
2

Enter course name:Arduino and Raspberry Pi

Enter course code:ECE312

Enter branch:ECE



Enter year:2

Operation complete

1.Cont 2.Exit:1

Enter type of user: 1.Admin 2.Student

2

Enter your branch:CS

Enter your year:1

OPEN ELECTIVES

ENG01 -- Creative Writing

HSC01 -- Home Science

BRANCH ELECTIVES for CS, year1

CS297 -- Python Programming

Operation complete

1.Cont 2.Exit:1

Enter type of user: 1.Admin 2.Student

2

Enter your branch:ECE

Enter your year:2

OPEN ELECTIVES

ENG01 -- Creative Writing

HSC01 -- Home Science

BRANCH ELECTIVES for ECE, year2

ECE312 -- Arduino and Raspberry Pi

Operation complete

1.Cont 2.Exit:2

Exiting PGM...

