Exercice-1

SVPL utilisez les deux fichiers de Dart pour cette exercice, un pour définir toutes les fonctions (une pour chaque sous-exercice), et l’autre pour tester les fonctions.

1.1

Écrivez un programme qui calcule les intérêts accumulés (intérêts composés) chaque année pendant 10 ans, par capitalisation d'une somme de 10000 $ placée en banque au taux fixe de 5,5 %.

import 'dart:math' as math;

main() {

int PV = 10000;

double interest = 0.055;

int period = 10;

double result = PV \* math.pow((1 + interest), period);

print(result);

}

1.2

Écrivez un programme qui calcule les 100 premiers termes de la table de multiplication par 14.

main() {

for(int i = 1; i <= 100; i++) {

print(i \* 14)

}

}

1.3

Écrivez un programme qui affiche la suite de symboles suivante :

\*

\*\*\*

\*\*\*\*\*

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

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

main() {

string result;

for(int i = 1; i<=9; i + 2) {

for(int j = 1; j < 9-i; j++) {

result = '$result ';

}

for(int k = 1; k < j; k++) {

result = '$result\*';

}

print(result);

result = '';

}

}

1.4

Écrivez un programme qui convertisse en degrés, minutes, secondes un angle fourni au départ en radians.

1.5

Écrivez un script qui compte le nombre d'occurrences du caractère « o » dans cette phrase.

1.6

Écrivez un script qui détermine si une chaîne de caractères donnée est un palindrome (c'est-à-dire une chaîne qui peut se lire indifféremment dans les deux sens), comme par exemple « radar » ou « s.o.s ».

1.7

Écrivez un programme qui analyse un par un tous les éléments d ’une liste de mots (par exemple : ['Jean', 'Maximilien', 'Brigitte', 'Sonia', 'Jean-Pierre', 'Sandra']) pour générer deux nouvelles listes. L’une contiendra les mots comportant moins de 6 caractères, l’autre les mots comportant 6 caractères ou davantage.

main() {

array liste = ['Jean', 'Maximilien', 'Brigitte', 'Sonia', 'Jean-Pierre', 'Sandra'];

array petiteliste;

array grandeliste;

foreach(var nom in liste){

if(nom.length > 6) {

grandeliste.append(nom);

} else {

petitelist.append(nom);

}

}

}

1.8

Convertir une note scolaire N quelconque en une note standardisée suivant le code ci-dessous :

A : N >= 90 %

B : 90 % > N >= 80 %

C : 80 % > N >= 70 %

D : 70 % > N >= 60 %

E : N < 60 %

main () {

int n = 75;

print(NoteStandardisee(n))

}

function NoteStandardisee (int note) {

switch() :

case(note >= 90) :

return 'A';

break;

case(note >= 80 && note < 90) :

return 'B';

break;

case(note >= 70 && note < 80) :

return 'C';

break;

case(note >= 60 && note < 70) :

return 'D';

break;

default :

return 'E';

break;

}

1.9

Soit la liste suivante : ['Jean-Michel', 'Marc', 'Vanessa', 'Anne', 'Maximilien', 'Alexandre-Benoît', 'Louise']. Écrivez un script qui affiche chacun de ces noms avec le nombre de caractères correspondant.

1.10

Définissez une fonction compteMots(ph) qui renvoie le nombre de mots contenus dans la phrase ph. On considère comme mots les ensembles de caractères inclus entre des espaces.