print(**"Ejercicio 1"**)

**class** Rectangulo():

**def** \_\_init\_\_(self,alt,base):

self.altura = alt

self.base = base

**def** devuelveArea(self):

print (**"El area del rectangulo es :"**,self.altura\*self.base)

print()

alt= int(input(**"Ingrese la altura del rectangulo:"**))

print()

base= int(input(**"Ingrese la base del rectangulo:"**))

print()

trinium =Rectangulo(alt, base)

trinium.devuelveArea()

print(**"Ejercicio 2"**)

**from** math **import** pi

**class** Circulo():

**def** \_\_init\_\_(self,rad):

self.radio=rad

**def** devuelvePerim(self):

print(**"El perimetro del circulo con radio"**,rad,**"es:"**,self.radio\*pi)

print()

**def** devuelveArea(self):

print(**"El area del circulo con radio"**,rad,**"es:"**,self.radio\*self.radio\*pi)

print()

rad= int(input(**"Ingrese radio del circulo:"**))

print()

cir = Circulo(rad)

cir.devuelveArea()

cir.devuelvePerim()

print(**"Ejercicio 3"**)

**import** random

**class** Persona():

**def** \_\_init\_\_(self, nombre,edad,peso,sexo,altura):

self.nombre=nombre

self.edad=edad

self.peso=peso

self.sexo=sexo

self.altura=altura

self.dni= self.generar\_dni()

**def** es\_mayor\_de\_edad(self,edad):

**if** (edad>=18):

print(**"Es mayor de edad"**)

**else** :

print(**"No es mayor de edad"**)

**def** print\_data(self,nombre,edad,peso,sexo,altura):

print (**"Nombre:"**,nombre)

print (**"Edad:"**,edad)

print (**"Peso"**,peso)

print (**"Sexo"**,sexo)

print (**"Altura"**,altura)

**def** generar\_dni(self):

**return** random.randint(0,99999999)

nombre=input(**"Ingrese el nombre:"**)

print()

edad=int(input(**"Ingrese la edad:"**))

print()

peso=int(input(**"Ingrese el peso:"**))

print()

sexo=input(**"Ingrese el sexo (F/M):"**)

print()

altura=float(input(**"Ingrese la altura:"**))

print()

per=Persona(nombre,edad,peso,sexo,altura)

per.es\_mayor\_de\_edad(edad)

print()

print (**"DNI:"**,per.dni)

print()

print(**"Ejercicio 4"**)

**class** Estudiante(Persona):

**def** \_\_init\_\_(self,nombre,edad,peso,sexo,altura, carrera, añoingreso, cant, aprobadas):

Persona.nombre=nombre

Persona.edad=edad

Persona.peso=peso

Persona.sexo=sexo

Persona.dni= per.dni

Persona.altura=altura

self.carrera=carrera

self.año=añoingreso

self.cant=cant

self.aprobadas= aprobadas

**def** avance(self):

porc=(self.aprobadas\*100)/self.cant

print(**"El porcentaje de materias de la carrera que tiene aprobadas es:"**,porc,**"%"** )

print()

**def** edad\_ingreso(self):

añoscursado= 2017-self.año

edad\_ingreso =self.edad - añoscursado

print(**"La edad que tenia al ingresar era:"**, edad\_ingreso, **"años"**)

print()

carrera= input(**"Ingrese carrera:"**)

print()

añoingreso=int(input(**"Ingrese año de ingreso:"**))

print()

cant= int(input(**"Ingrese cantidad de materias de la carrera:"**))

print()

aprobadas = int(input(**"Ingrese cantidad de materias aprobadas:"**))

print()

nuevo = Estudiante(nombre,edad,peso,sexo,altura, carrera, añoingreso, cant, aprobadas)

nuevo.avance()

nuevo.edad\_ingreso()

print(**"Ejercicio 5"**)

print(**"Escribir una funcion que tome como parámetro una lista de Estudiantes, y devuelva un diccionario"**

**"con las carreras como keys, y la cantidad de estudiantes en cada una de ellas como values"**)

print()

**def** diccionario(lista):

dicc = {}

**for** est **in** lista:

**if** est.carrera **in** dicc.keys():

dicc[est.carrera] += 1

**else**:

dicc[est.carrera] = 1

**return** dicc

e1 = Estudiante(**"Jorge"**, 21, **"M"**, 60, 1.56, **"ISI"**, 2015, 40, 20)

e2 = Estudiante(**"Mariana"**, 35, **"F"**, 60, 1.56, **"IQ"**, 2016, 50, 10)

e3 = Estudiante(**"Sergio"**, 23, **"M"**, 60, 1.56, **"ISI"**, 2011, 30, 15)

e4 = Estudiante(**"Richard"**, 54, **"M"**, 45, 54, **"IM"**, 2001, 12, 4)

lista = [e1,e2,e3,e4]

print(diccionario(lista))

##################################################################

##################################################################

#

#Una forma “linda” de hacer el metodo avance del ejercicio 4 es:

# **def** avance(self):

print(**"El avance en la carrera de {0} es {1}%"**.format(Persona.nombre, round(self.aprobadas \* 100 / self.cant,2)))

#

##################################################################

##################################################################