Object Oriented Programming 1 - Assignment

a) ¶

Create a class `Person` which contains 2 properties, `_name` and `_age`.

- Implement its initializer method which initialize `_name` and `_age`.
- Implement its `__str__()` method which return string in the format of `"Person: name={_name}, age={_age}"`
- Define a class variable `MIN_ADULT_AGE` with value `18`.
- Implement an instance method `is_adult()` which returns `True` if the age is equal or above `MIN_ADULT_AGE`, else returns `False`.

Sample Output:

```
Person: name=Alan, age=20
Is adult: True
```

In [3]:

```
class Person:

    MIN_AULDT_AGE = 18

    def __init__(self, name, age):
        self._name = name
        self._age = age

    def __str__(self):
        return '{}: name={}, age={}'.format(self.__class__.__name__, self._name, self._
age)

    def is_adult(self):
        return self._age >= Person.MIN_AULDT_AGE


p = Person('Alan', 20)
print(p)
print('Is adult:', p.is_adult())
```

```
Person: name=Alan, age=20
Is adult: True
```

b)

Create a class `Shape` which contains following attributes:

- Define a class variable `PI` with value `3.14` .
- Define a class method `area_circle()` which takes in a `radius` value and returns area of the circle.
- Define a static method `area_rectangle()` which takes in `width` and `length` values, and returns area of the rectangle.

Sample Output:

```
Area of Circle: 12.56
Area of Rectangle: 8
```

In [7]:

```
class Shape:

    PI = 3.14

    @classmethod
    def area_circle(cls, radius):
        return Shape.PI * (radius**2)

    @staticmethod
    def area_rectangle(width, height):
        return width * height


print('Area of Circle:', Shape.area_circle(2))
print('Area of Rectangle:', Shape.area_rectangle(2,4))
```

```
Area of Circle: 12.56
Area of Rectangle: 8
```