What is the difference between a class and an instance of a class? X

In a Python class, how do you hide an attribute from code outside the class? X

Provide code examples demonstrating the usage of classes in Python. X

Hello,

Classes are objects composed of data and functionality. An instance of a class is an object of a given type (class) with its own unique state.

To hide an attribute from outside of a class, you can restrict access by naming an attribute with two leading underscores.

Library Patron Class

```python
class Patron:
    # class attribute. accessible to all instances of the class.
    CHECKOUT_LIMIT = 5

    # instance constructor. the self argument refers to the instance.
    def __init__(self, name, age = None):
        # instance attribute. data only accessible to instance.
        self.name = name
        self.inventory = dict()

        # private
        self.__age = age

    # private attribute getter
    def get_age(self):
        return self.__age

    # instance property
    @property
    def info(self):
        print(f"Name: {self.name}")
        print(f"\nAge: {self.get_age()}")
        print("\nItems Checked Out:")
        for item in self.inventory:
            print(f"{item}")

    # instance method
    def checkout_items(self, items):
        self.inventory.update(items)
        return self.inventory

    def return_items(self, items):
      for item in items:
        del self.inventory[item]
      return self.inventory

    # class method. the cls argument refers to the class.
    @classmethod
    def get_checkout_limit(cls):
        print(f"\nThe checkout limit is {cls.CHECKOUT_LIMIT} items maximum.")

    # static method (called by the class)
    @staticmethod
    def show_receipt(items):
        print("You have these items:\n")
        for item in items:
            print(f"{item}")
        print("\nItems must be returned within 3 weeks of checkout.")
```

```python
# create instance of type Patron
lily = Patron('Lily')
print(type(lily))
print(lily.name)
```

```
<class '__main__.Patron'>
Lily
```

```
Patron.CHECKOUT_LIMIT == lily.CHECKOUT_LIMIT
```

True

```
Patron.get_checkout_limit()
```

The checkout limit is 5 items maximum.

```
james = Patron('James', 80)
print('Age attribute public?', hasattr(james, '__age'))
print(f"Patron's age: {james.get_age()}")
```

Age attribute public? False
Patron's age: 80

```
items_out = james.checkout_items({'Jane Eyre': 'Movie', 'Emma': 'Book', 'Little Women': 'Book'})
Patron.show_receipt(items_out)
```

You have these items:

Jane Eyre
Emma
Little Women

Items must be returned within 3 weeks of checkout.

```
james.return_items({'Emma': 'Book'})
```

{'Jane Eyre': 'Movie', 'Little Women': 'Book'}

```
james.info
```

Name: James

Age: 80

Items Checked Out:
Jane Eyre
Little Women

Thanks,

Lauren

References

Geeks for Geeks. (2021, January 4). *Python Classes and Objects*.

https://www.geeksforgeeks.org/python-classes-and-objects/

The Python Language Reference. (n.d.). *Classes.*

https://docs.python.org/3/tutorial/classes.html