Four Pillars of OOP

- Abdullah Zayed

contact@zayedabdullah.com

Agenda

Inheritance

Access Modifiers

Polymorphism

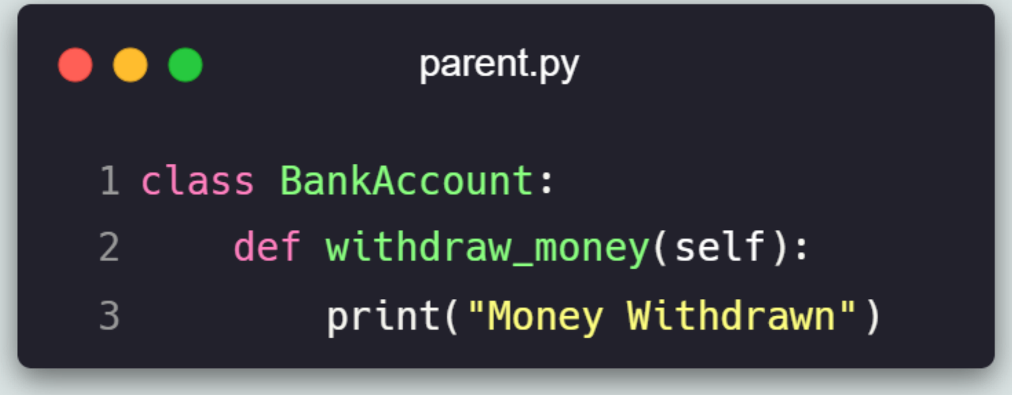
Abstract Class

Inheritance

INHERIT PROPERTIES

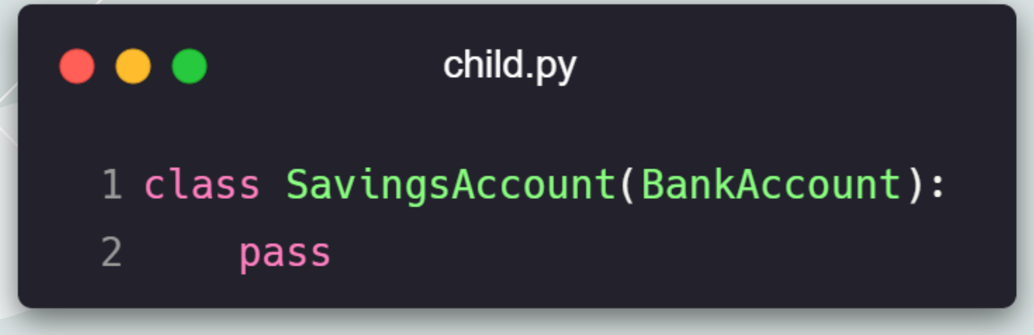
Inheritance

The child gets a copy of everything the parent offers.



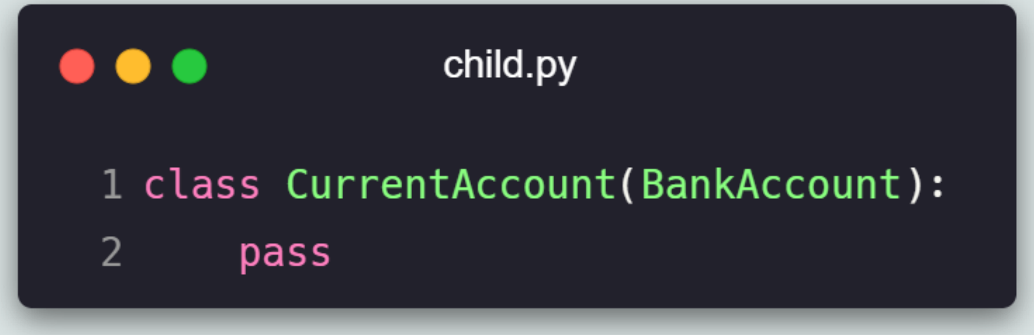
parent.py

```python
class BankAccount:
    def withdraw_money(self):
        print("Money Withdrawn")
```



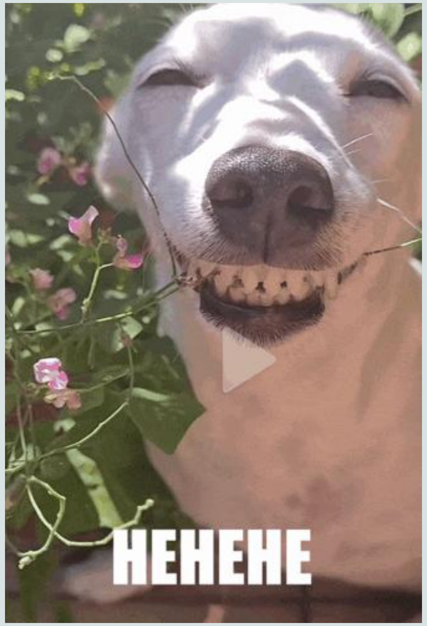
child.py

```python
class SavingsAccount(BankAccount):
    pass
```



child.py

```python
class CurrentAccount(BankAccount):
    pass
```



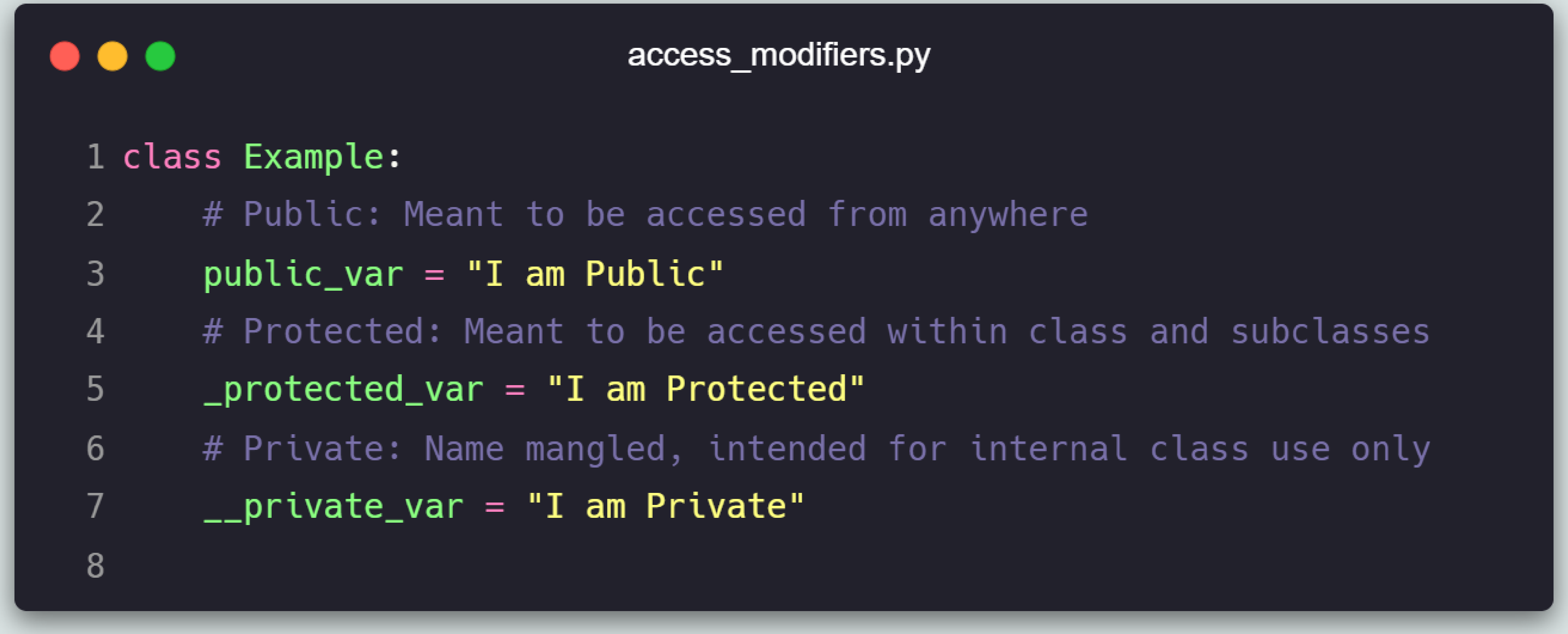
Access Modifiers

WHO CAN ACCESS WHAT 😉

contact@zayedabdullah.com

Access Modifiers

Just a **hint** of who can access what…



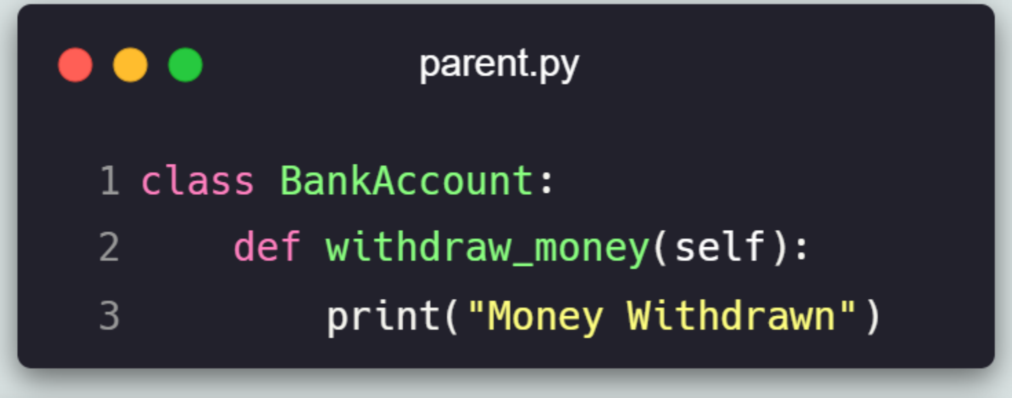
```python
class Example:
    # Public: Meant to be accessed from anywhere
    public_var = "I am Public"
    # Protected: Meant to be accessed within class and subclasses
    _protected_var = "I am Protected"
    # Private: Name mangled, intended for internal class use only
    __private_var = "I am Private"

```

Polymorphism

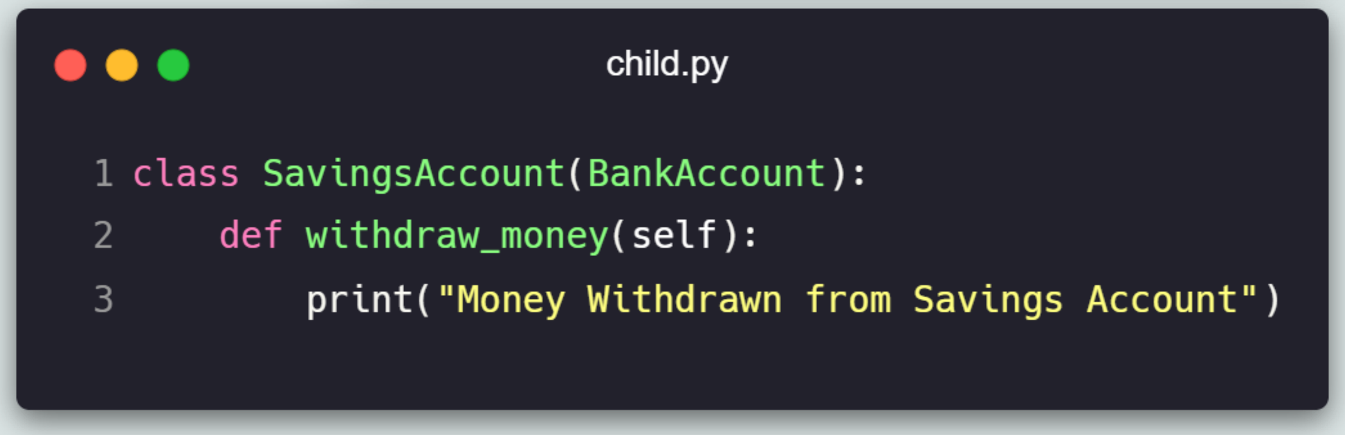
INHERIT PROPERTIES & REPLACE SOMETHING 😏

Polymorphism

The child replaces some properties inherited from the parent.



```python
class BankAccount:
    def withdraw_money(self):
        print("Money Withdrawn")
```



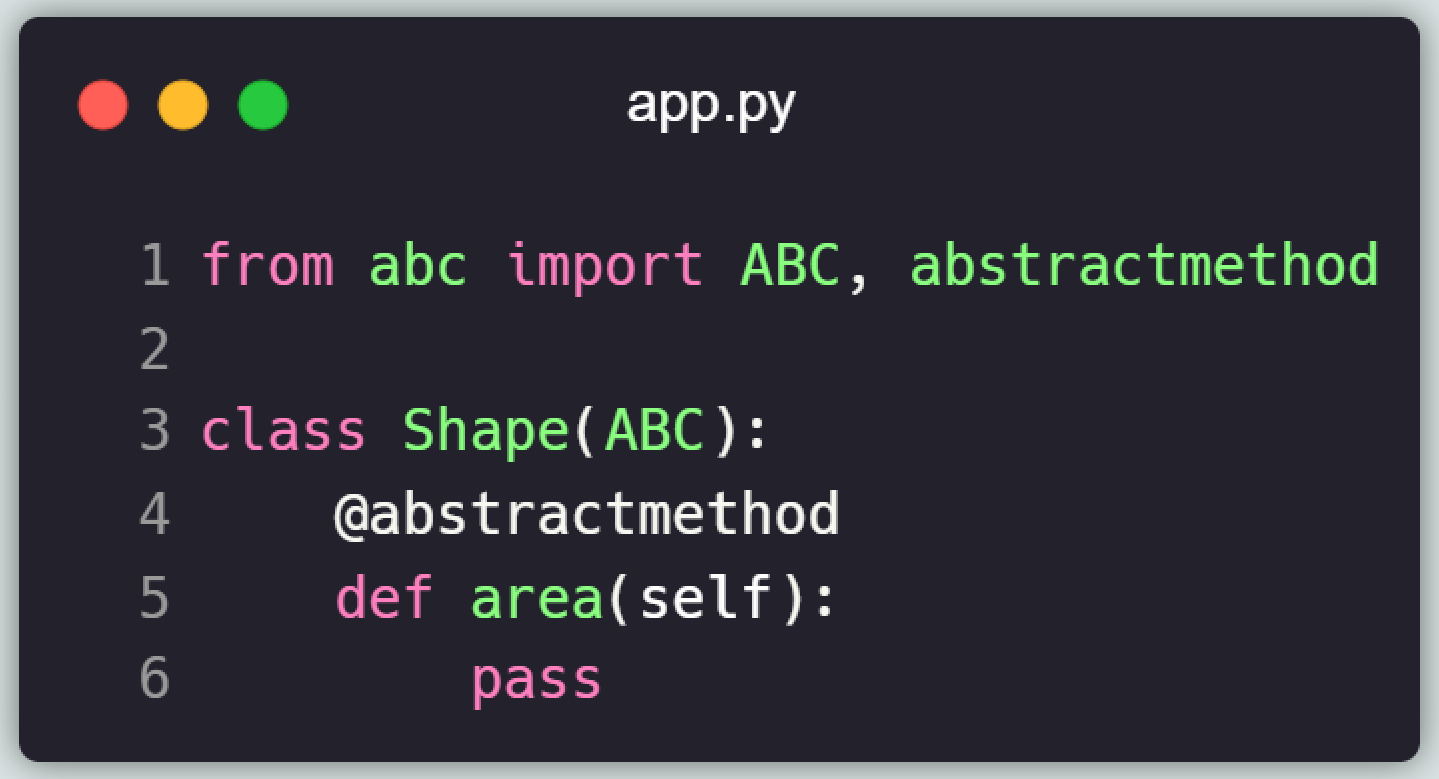
```python
class SavingsAccount(BankAccount):
    def withdraw_money(self):
        print("Money Withdrawn from Savings Account")
```

Abstract Class

WHAT TO BUILD 😳

Abstract Class

Just a **hint** of what to build…



app.py

```python
from abc import ABC, abstractmethod

class Shape(ABC):
    @abstractmethod
    def area(self):
        pass
```

Thank you

Abdullah Zayed

contact@zayedabdullah.com

www.zayedabdullah.com

linkedin.com/in/abdullahzayed01/