Lecture

Working with Objects

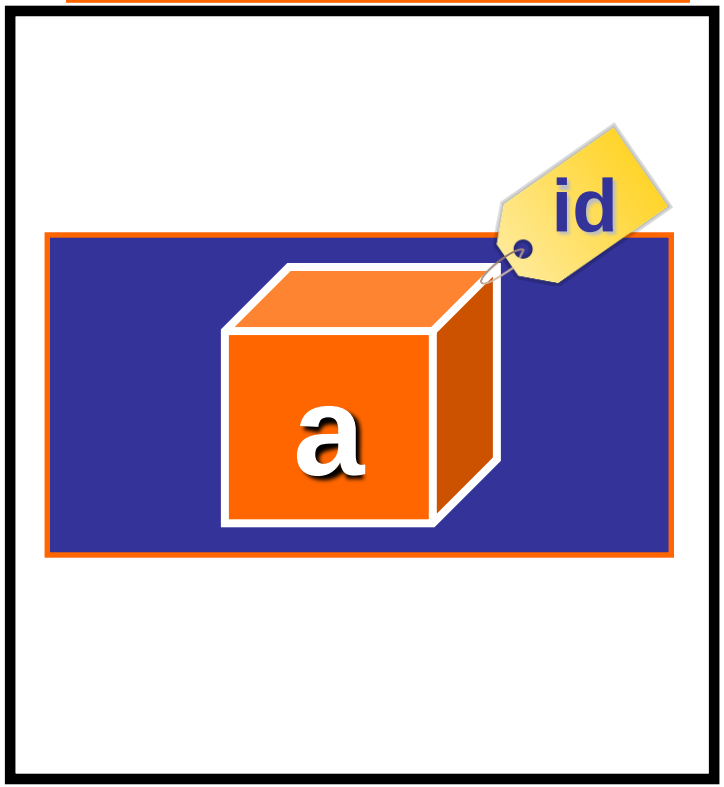
Objects are passed by reference

Working with objects

```
>>> a = [6, 2, 8, 2]
>>> print_data(a)
```

```
def print_data(lst):
    for elem in lst:
        print(elem)
```

Memory



Working with objects

```
>>> a = [6, 2, 8, 2]
>>> print_data(a)
```

```
def print_data(lst):
    for elem in lst:
        print(elem)
```

Memory

Working with objects

```
>>> a = [6, 2, 8, 2]
>>> print_data(a)
```

```
def print_data(lst):
    for elem in lst:
        print(elem)
```

Memory

Working with objects

```
>>> a = [6, 2, 8, 2]
>>> print_data(a)
```

Pass a reference

```
def print_data(lst):
    for elem in lst:
        print(elem)
```

Memory

Working with objects

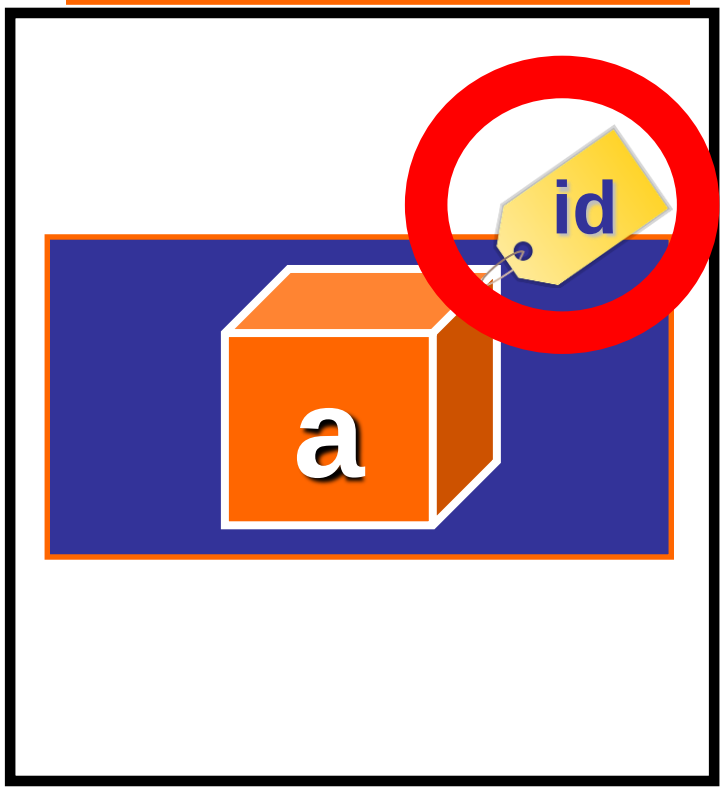
```
>>> a = [6, 2, 8]
>>> print_data(a)
```

id

Pass a reference

```
def print_data(lst):
    for elem in lst:
        print(elem)
```

Memory



Working with objects

```
>>> def print_data(lst):
        for elem in lst:
            print(elem)


>>> a = [6, 2, 8, 2]
>>> print_data(a)
6
2
8
2
```

Working with objects

```
>>> class Child:
    def __init__(self, name, age):
        self.name = name
        self.age = age


>>> def update_age(lst):
    print("==== Updating Age ====")
    print("\n-> Initial list:")
    for child in lst:
        print(f"Name: {child.name}; Age: {child.age}")

    # Update age
    for child in lst:
        child.age += 1

    print("\n-> Final list:")
    for child in lst:
        print(f"Name: {child.name}; Age: {child.age}")


>>> classroom = [Child("Nora", 10), Child("Daniel", 13), Child("Jack", 7)]
>>> update_age(classroom)
```


Working with objects

```
>>> class Child:
    def __init__(self, name, age):
        self.name = name
        self.age = age


>>> def update_age(lst):
    print("==== Updating Age ====")
    print("\n-> Initial list:")
    for child in lst:
        print(f"Name: {child.name}; Age: {child.age}")

    # Update age
    for child in lst:
        child.age += 1

    print("\n-> Final list:")
    for child in lst:
        print(f"Name: {child.name}; Age: {child.age}")


>>> classroom = [Child("Nora", 10), Child("Daniel", 13), Child("Jack", 7)]
>>> update_age(classroom)
```

Working with objects

```
>>> class Child:
    def __init__(self, name, age):
        self.name = name
        self.age = age
```

```
>>> def update_age(lst):
    print("==== Updating Age ====")
    print("\n-> Initial list:")
    for child in lst:
        print(f"Name: {child.name}; Age: {child.age}")

    # Update age
    for child in lst:
        child.age += 1

    print("\n-> Final list:")
    for child in lst:
        print(f"Name: {child.name}; Age: {child.age}")
```

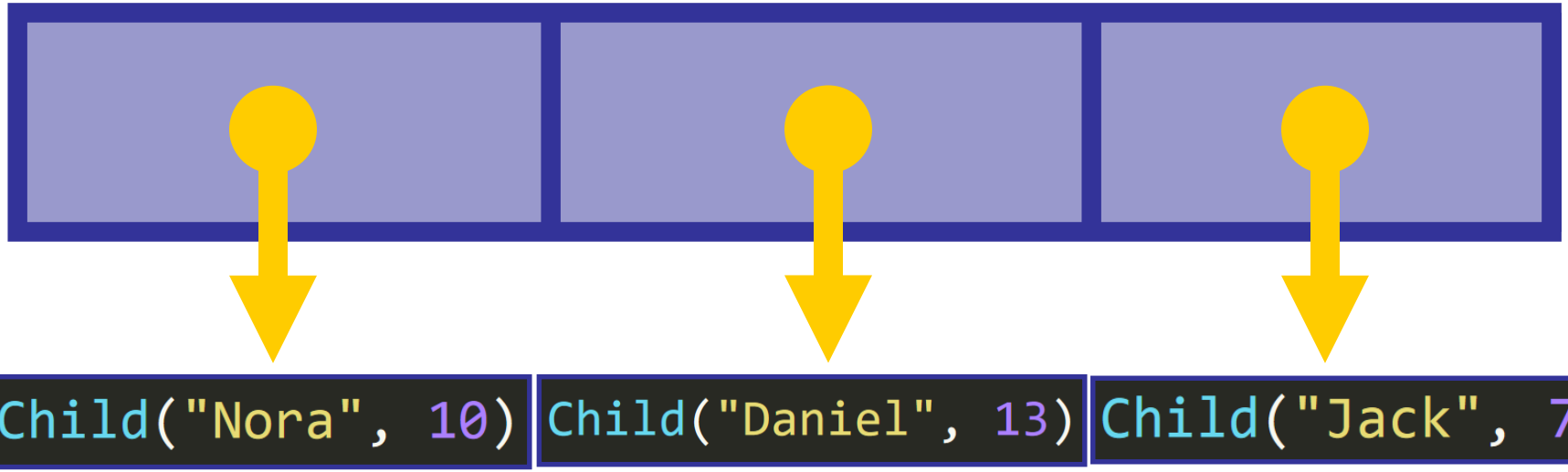
```
>>> classroom = [Child("Nora", 10), Child("Daniel", 13), Child("Jack", 7)]
>>> update_age(classroom)
```

Working with objects

```
>>> class Child:
    def __init__(self, name, age):
        self.name = name
        self.age = age


>>> def update_age(lst):
    print("==== Updating Age ====")
    print("\n-> Initial list:")
    for child in lst:
        print(f"Name: {child.name}; Age: {child.age}")

    # Update age
    for child in lst:
        child.age += 1

    print("\n-> Final list:")
    for child in lst:
        print(f"Name: {child.name}; Age: {child.age}")
```

```
>>> classroom = [Child("Nora", 10), Child("Daniel", 13), Child("Jack", 7)]
>>> update_age(classroom)
```

Working with objects



Working with objects

```
for child in lst:
    child.age += 1
```

```
Child("Nora", 10)
```

```
Child("Daniel", 13)
```

```
Child("Jack", 7)
```

Working with objects

Reference

```
for child in lst:
    child.age += 1
```

```
Child("Nora", 10)
```

```
Child("Daniel", 13)
```

```
Child("Jack", 7)
```

Objects are passed by reference

Time to Practice