Python - Lists

What's a list?

In some languages you would do this stuff with ARRAYS, which work in a very similar way

A list is a collection of something, like integers:

```
numbers = [3,5,9,6]
print(numbers)
print(numbers[0])
print(numbers[1])
print(numbers[2])
print(numbers[3])
```

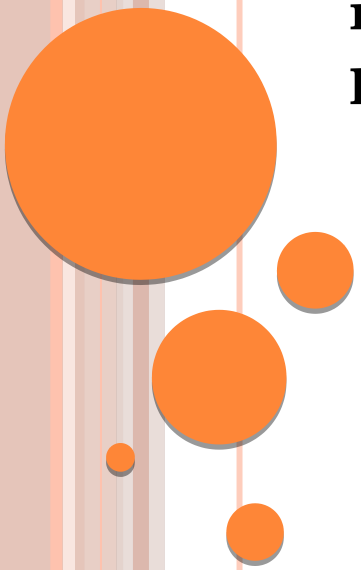
You use square brackets to write the list and then square brackets again to say which LIST ELEMENT you want. We start counting from 0 so the first number in our list is numbers[0] and the last one is numbers[3].

Python - Lists

Editing Lists

As well as reading individual elements we can write to individual elements too:

```
numbers = [3,5,9,6]
print(numbers)
numbers[1] = 2
print(numbers)
```



Python - Lists

Lists of items

Lists are really good when you use two or more together. Lets say we want to store a load of people's ages:

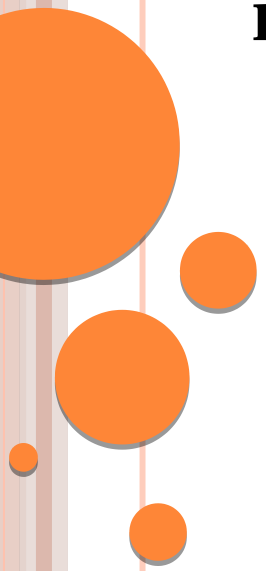
```
names = ["Paul","Phillip","Paula","Phillipa"]
ages = [12,15,11,14]

print(names[0],"is",ages[0])
print(names[1],"is",ages[1])
print(names[2],"is",ages[2])
print(names[3],"is",ages[3])
```

Python - Lists

Strings

Strings are really lists of characters.
As such you can do the following:

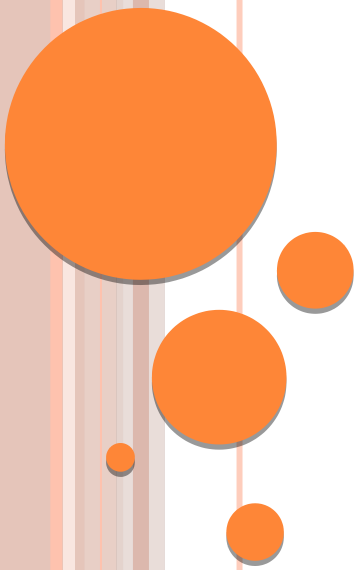
name = “alex”
print(name[2])



Python - Lists

If you find your list is too short, you can add one more value at a time by using the APPEND procedure.

```
names = [“Rita”, “Sue”]
names.append(“Bob”)
print(names)
```

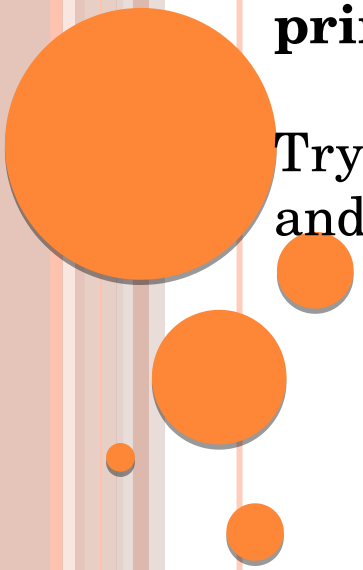


Python - Lists

If you want to add more than one value at a time you can use EXTEND procedure.

```
names = [“Graham”,”Eric”,”Terry G.”]
extraNames = [“Terry J.”, “John”, “Michael”]
names.extend(extraNames)
print(names)
```

Try the code above with APPEND instead of EXTEND and see what happens.



Python - Lists

You can search a list using IN:

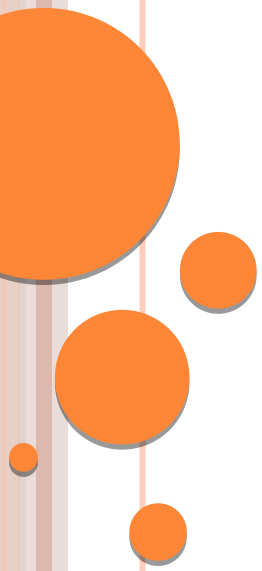
```
names = [“Graham”,”Eric”,”Terry”]

if “John” in names:
    print(“John present”)
if “Eric” in names:
    print(“Eric present”)
```

Try the code above

Python - Lists

If you want to know where in the list it is, you can use INDEX:

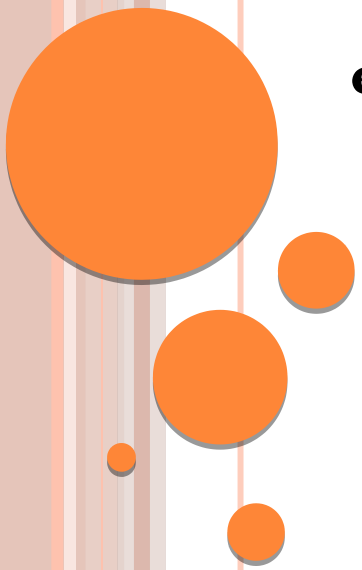
```
names = [“Graham”,”Eric”,”Terry”]
position = names.index(“John”)
print(position)
```



Python - Lists

BUT be careful because if the thing you're searching for isn't there, you'll get a nasty error. Perhaps try:

```
names = ["Graham","Eric","Terry"]

if "Eric" in names:
    position = names.index("Eric")
    print("Eric in position",position)
else:
    print("Eric not found")
```



Python - Lists

For loops often go hand in hand with Lists. Try the following code:

```
names = [“”]*5

for i in range(5):
        prompt = “Please enter in name “ + str(i+1) + ”: ”
        names[i] = input(prompt)

print(“Your names are: “)
for i in range (5):
        print(names[i])
```