Last time – getting user input

- This is an assignment statement.
- name is a variable.
- The value of:

  ```python
  raw_input("What is your name? ")
  ```

  will be whatever the user types as input.

Enter to exit

- Many examples in the book end with:

  ```python
  raw_input("Press enter to exit")
  ```

- Python waits for the user to enter something. As soon as the user hits the enter key, the program is done and it exits.
- Notice the value produced by `raw_input()` is not put into a variable; it is just thrown away.
The empty string

- When the user just types the Enter key, `raw_input()` produces a string, but it is a string containing no characters.
- It is called the empty string.
- The length of string “Daniel” is 6
- The length of the empty string “” is 0

Checking for the empty string

- Use the if statement:

```python
if name == "" :
    print "No name received."
```

The condition

- The condition (for example, name == "") is testing for equal values.
- It is not an assignment statement!
- It uses the operator ==, which is different from the = in an assignment statement.

Syntax details

- The condition is followed by a colon (the : thing).
- The statements that are done if the condition is true are indented.
Block

- You can have lots of statements under an if statement.

```python
if name == "":
    print "No name received."
    print "We will not proceed."
    raw_input("Press enter to exit: ")
```

Block

- The indented statements form a block. A block is always done together, or skipped together.

```python
if name == "":
    print "No name received."
    print "We will not proceed."
    raw_input("Press enter to exit: ")
```

If ... else...

```python
if name == "":
    print "No name received."
else:
    print "You are the", name+"meister"
```

Either the block under if is done, or the block under else.
One or the other is executed.

If ... else...

```python
if name == "":
    print "No name received."
else:
    print "You are the", name+"meister!"
```

The command else is NOT indented.
In IDLE, use backspace key to get back to left side.
If … else…

```python
if name != "":
    print "You are the",name+"meister!"
```

- The symbol != means “is not equal”
- The symbol == means “is equal”

if … elif…else

```python
if name == 'Fido':
    species = 'dog'
elif name == 'Fluffy':
    species = 'cat'
else:
    species = 'human'
```

- Only one of the blocks gets executed.
- Tries first condition; if that works, execute that block.
- If not, try second condition.
- If that fails too, the else gets executed.
- Can have as many elif’s as you want.

How to use in quiz?

- Use raw_input() to get the user’s answers.
- Use if-elif-else to check for the right answer.
- How to keep score?
- You’ll probably need a variable to remember the score.