“Guess the Number”

Invent Your Own Computer Games with Python

Taesoo Kwon
Heejin Park

College of Information and Communications
Hanyang University

Introduction (1/2)

- **The "Guess the Number" Game**
- **Code Explanation**
  - Arguments
  - Blocks
  - Conditions and Booleans
  - `if` statements

The "Guess the Number" Game

"Guess the Number" Game

- **Computer** will think of a **random number from 1 to 20.**
- Ask you to **guess the number.**
  - You only get **six guesses.**
  - but the computer will tell you if your guess is **too high or too low.**
- If you guess the number **within six tries, you win.**

The "Guess the Number" Game

- Sample Run of "Guess the Number"

```
Hello! What is your name?
Albert
Well, Albert, I am thinking of a number between 1 and 20.
Take a guess.
10
Your guess is too high.
Take a guess.
2
Your guess is too low.
Take a guess.
4
Good job, Albert! You guessed my number in 3 guesses!
```

Building Blocks

- **The `random.randint()` Function**

```
9. number = random.randint(1, 20)
```

  - store the return value in a variable named `number`.
  - **`randint()`** function is provided by the random module.
    - » we precede it with **random.**
    - » will return a random integer.
      - between the **two integers** we give it. (separated by a comma)
      - here, It should return an integer between **1 and 20**.

Building Blocks

- Type **`import random`** to import the random module.

```
>>> import random
>>> random.randint(1, 20)
12
>>> random.randint(1, 20)
18
>>> random.randint(1, 20)
3
>>> random.randint(1, 20)
18
>>> random.randint(1, 20)
7
```

```
>>> random.randint(1, 4)
3
>>> random.randint(1, 4)
4
>>> random.randint(1000, 2000)
1294
>>> random.randint(1000, 2000)
1585
```

Building Blocks

- **`While` statement**

```
while guessesTaken < 6:
```

  - Is made up of the **`while` keyword**, followed by **an expression**, followed by **a colon(the : sign).**

- **Condition**
  - **The expression** next to the while keyword is also called a condition.

Building Blocks

`if` statements

- works almost the same way as a `while` statement.
- But **unlike** the `while`-block
  - It just continues on down to the next line.
  - In other words, if statements won't loop.

```
if fizzy < 10:    while fizzy > 6:
```

if keyword — condition

while keyword — condition

The "Guess the Number" Game

Guess the Number's Source Code

```
# This is a guess the number game.
import random

guessesTaken = 0

print 'Hello! What is your name?'
myName = raw_input()

number = random.randint(1, 20)
print 'Well, ' + myName + ', I am thinking of a number between 1 and 20.'

while guessesTaken < 6:
    print 'Take a guess.' # There are four spaces in front of print.
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1

    if guess < number:
        print 'Your guess is too low.' # There are eight spaces in front of print.

    if guess > number:
        print'Your guess is too high.'

    if guess == number:
        break

if guess == number:
    guessesTaken = str(guessesTaken)
    print 'Good job, ' + myName + '! You guessed my number in ' + guessesTaken + ' guesses!'

if guess != number:
    number = str(number)
    print 'Nope. The number I was thinking of was ' + number
```

Introduction (2/2)

- **Code Explanation - Step by step**
  - Make Minor Changes
  - What Exactly is Programming?
  - A Web Page for Program Tracing
- **Topics Covered In This Chapter**

The "Guess the Number" Game

- Guess the Number's Source Code

```
1. # This is a guess the number game.
2. import random
3.
4. guessesTaken = 0
5.
6. print 'Hello! What is your name?'
7. myName = raw_input()
8.
9. number = random.randint(1, 20)
10. print 'Well, ' + myName + ', I am thinking of a number between
    1 and 20.'
11.
12. while guessesTaken < 6:
13.     print 'Take a guess.' # There are four spaces in front of
                                print.
14.     guess = raw_input()
15.     guess = int(guess)
16.
17.     guessesTaken = guessesTaken + 1
```

The “Guess the Number” Game

- Guess the Number's Source Code

```
18.
19.     if guess < number:
20.         print 'Your guess is too low.‘
21.
22.     if guess > number:
23.         print 'Your guess is too high.'
24.
25.     if guess == number:
26.         break
27.
28. if guess == number:
29.     guessesTaken = str(guessesTaken)
30.    print 'Good job, ' + myName + '! You guessed my number in '
    +  guessesTaken + ' guesses!'
31.
32. if guess != number:
33.     number = str(number)
34.     print 'Nope. The number I was thinking of was ' + number
```

Code Explanation

- **Comment**
  - Just reminds us what this program does.

```
1. # This is a guess the number game.
```

- **Modules**
  - Other programs that contain other functions that we can use.

- **`Import` statement**
  - Will **add modules** and **their functions** to our program.
  - Is made up of the **`import` keyword** followed by the **module name.**

```
2. import random
```

Code Explanation

- This creates a **new variable**
  - We will store the integer 0 here.

```
4. guessesTaken = 0
```

- These two lines are **identical to Hello World program**.

```
6. print 'Hello! What is your name?'
7. myName = raw_input()
```

Code Explanation

- We can **change the game's code** slightly.

```
9. number = random.randint(1, 20)
10. print 'Well, ' + myName + ', I am thinking of a number
    between 1 and 20.'
```

```
9. number = random.randint(1, 100)
10. print 'Well, ' + myName + ', I am thinking of a number
    between 1 and 100.'
```

Code Explanation

- Quiz

```
>>> randint(1, 20)
>>>
```

```
>>> random.randint(100, 100)
>>>
```

```
>>> random.randint(5.0, 10.0)
>>>
```

```
>>> random.randint(5.5, 10.0)
>>>
```

Code Explanation

Arguments

- The values that are passed to a function when the function is called.

```
raw_input()
random.randint(1, 20)
```

  - The **`raw_input()`** function has **no arguments.**
  - The **`randint()`** function call has **two arguments.**
    - » Programmers say that the arguments are **delimited** by commas.

Code Explanation

- Quiz

```
>>> random.randint(1)
>>>
```

```
>>> random.randint(1, 2, 3)
>>>
```

Code Explanation

- **`Print` statement**

```
print 'Well, ' + myName + ', I am thinking of a
number between 1 and 20.'
```

  - The **plus signs** concatenate the three strings.
  - The **commas are *inside* the quotes,** and part of the strings themselves.

Code Explanation

- **`While` statement**

```
while guessesTaken < 6:
```

  - Is made up of the **`while` keyword**, followed by **an expression**, followed by **a colon(the : sign).**

- **Condition**
  - **The expression** next to the while keyword is also called a condition.

Code Explanation

Blocks

- A block is made up of several lines of code grouped together.

```
while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1

    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'
```

Code Explanation

Blocks

- where a **block begins and ends** by looking at the line's **indentation.**

```
while guessesTaken < 6:
....print 'Take a guess.'
....guess = raw_input()
....guess = int(guess)

....guessesTaken = guessesTaken + 1

....if guess < number:
........print 'Your guess is too low.'

....if guess > number:
........print 'Your guess is too high.'
```

Code Explanation

Loop block

- Call the block after the `while` keyword a **loop block.**
  - also called a **while-block.**
  - If the condition is **true**
    - » Program enters the **loop block again.**
  - If the condition is **false**
    - » Program **jumps** down to the line **after the loop block.**

Code Explanation

Conditions and Booleans

```
while guessesTaken < 6:
```

- Called the expression that came after the `while` keyword the **condition.**
- It contains **two values** connected by **an operator**
  - » **Two values**
    : variable `guessesTaken`, integer value 6
  - » **Operator**
    : the < sign, which is called the **"less than"** sign.

Code Explanation

Conditions and Booleans

- **Comparison operators.**

Operator Sign	Operator Name
<	**Less** than
>	**Greater** than
<=	**Less** than or **equal** to
>=	**Greater** than or **equal** to
==	**Equal** to
!=	**Not equal** to

Code Explanation

Conditions and Booleans

- **Boolean**

```
True
False
```

  - There are two and only **two values.**
  - Must type **`True`** of **`False`**(not **`true`** or **`fALSe`**).

- **Condition**
  - An expression that uses **comparison operators.**
  - Will always evaluate to a **boolean value.**

Code Explanation

- Type in the following conditions.

```
>>> 0 < 6
True
>>> 6 < 0
False
>>> 50 < 10
False
>>> 10 < 11
True
>>> 10 < 10
False
```

Code Explanation

- Quiz

```
>>> 0 > 6
>>> 6 > 0
>>> 10 > 10
>>> 10 == 10
>>> 10 == 11
>>> 11 == 10
>>> 10 != 10
>>> 10 != 11
>>> 'Hello' == 'Hello'
>>> 'Hello' == 'Good bye’
>>> 'Hello' == 'HELLO'
>>> 'Good bye' != 'Hello'
```

Code Explanation

- **Looping with `while` statements**
  - The while statement marks the beginning of a **loop.**
  - If the condition evaluates to **True**
    - » the execution **moves inside the while-block.**
  - If the condition evaluates to **False**
    - » the execution **moves all the way past the while-block.**

Code Explanation

- Looping with `while` statements

```
12. while guessesTaken < 6:
13.     print 'Take a guess.'
14.     guess = raw_input()
15.     guess = int(guess)
16.
17.     guessesTaken = guessesTaken + 1
18.
19.     if guess < number:
20.     print 'Your guess is too low.'
21.
22.     if guess > number:
23.     print 'Your guess is too high.'
24.
25.     if guess == number:
26.         break
27.
28. if guess == number:
```

If True… …go inside the loop-block to here.

If False… …go past the loop-block to here.

Code Explanation

- **The Player Guesses**
  - The program now asks us for a guess.
  - We store this guess in a variable named **`guess.`**

```
13.     print 'Take a guess.'
14.     guess = raw_input()
```

Code Explanation

- **`int()` Function**

```
15.     guess = int(guess)
```

  - Converting **Strings to Integers.**
  - The **`raw_input()`** function returned a string of text that player typed.
    - » But in our program, we will **want an integer.**
  - If the player enters 5 as their guess,
    - » will return the **string value '5'** and not the integer value 5.

Code Explanation

- Quiz

```
>>> int('42')
>>> int(42)
>>> int('hello')
>>> int('forty-two')
>>> int(' 42 ')
>>> 2 + int('2')
```

Code Explanation

- **Incrementing Variables**

```
17.  guessesTaken = guessesTaken + 1
```

  - The first time that we enter the loop block
    - » `guessesTaken` has the value of 0.
    - » take this value and **add 1** to it (0 + 1 is 1).
    - » **store the new value of 1** to `guessesTaken`.

  - When we **subtract** one from a value
    - » we are **decrementing** the value.

Code Explanation

`if` statements

- works almost the same way as a `while` statement.
- But **unlike** the `while`-block
  - It just continues on down to the next line.
  - In other words, if statements won't loop.

```
if fizzy < 10:    while fizzy > 6:
```

if keyword — condition

while keyword — condition

Code Explanation

`if` statements

- **Is the Player's Guess Too Low?**

```
19.     if guess < number:
20.         print 'Your guess is too low.'
```

  - If the condition evaluates to **True**
    - » then the code in the **if-block is executed.**
  - If the condition is **False**
    - » then the code in the **if-block is skipped.**

Code Explanation

`if` statements

- **Is the Player's Guess Too High?**

```
22.     if guess > number:
23.         print 'Your guess is too high.'
```

  - If the player's guess is **larger than the random integer**
    - » we **enter the `if`-block** that follows the `if` statement.
    - » The `print`line tells the player that their guess is too big.

Code Explanation

- **`break` Statement**

```
25.     if guess == number:
26.         break
```

  - if the **guess is equal to the random integer**
    - » we enter line 26, the `if`-block that follows it.

  - does not bother re-checking the `while` loop's condition.
    - » it just **breaks out immediately.**
    - » just the `break` keyword by itself, with no condition or colon.

Code Explanation

- **Check if the Player Won**

```
28. if guess == number:
29.     guessesTaken = str(guessesTaken)
30.     print 'Good job, ' + myName + '! You guessed
    my number in ' + guessesTaken + ' guesses!'
```

  - player **correctly guessed** the computer's number.

  - **function `str()`**
    - » how many guesses it took them.
    - » change the `guessesTaken` value into a string.

Code Explanation

- **Check if the Player Lost**

```
32. if guess != number:
33.      number = str(number)
34.      print 'Nope. The number I was thinking of was
'
    + number
```

  - player **failed to guess** correctly.

  - `str(number)`

    » inside the if-block, and only executes if the condition was `True`.

  - we have reached the **end of the code**, and the program **terminates.**

Code Explanation – step by step

- **Tracing through the program.**
  - Let's go over the code one more time.
  - To help you understand everything.
  - Remember what the **values of variables** are ourselves.

Code Explanation – step by step

```
# This is a guess the number game.
import random

guessesTaken = 0

print'Hello! What is your name?'
myName = raw_input()

number = random.randint(1, 20)
print'Well, ' + myName + ', I am thinking of a number between 1 and 20.'
```

Code Explanation – step by step

```
# This is a guess the number game.
import random

guessesTaken = 0

print'Hello! What is your name?'
myName = raw_input()

number = random.randint(1, 20)
print'Well, ' + myName + ', I am thinking of a number between 1 and 20.'
```

Code Explanation – step by step

```
# This is a guess the number game.
import random

guessesTaken = 0

print'Hello! What is your name?'
myName = raw_input()

number = random.randint(1, 20)
print'Well, ' + myName + ', I am thinking of a number between 1 and 20.'
```

Code Explanation – step by step

guessesTaken	0

```
# This is a guess the number game.
import random

guessesTaken = 0

print'Hello! What is your name?'
myName = raw_input()

number = random.randint(1, 20)
print'Well, ' + myName + ', I am thinking of a number between 1 and 20.'
```

Code Explanation – step by step

guessesTaken	0

```
# This is a guess the number game.
import random

guessesTaken = 0

print'Hello! What is your name?'
myName = raw_input()

number = random.randint(1, 20)
print'Well, ' + myName + ', I am thinking of a number between 1 and 20.'
```

Code Explanation – step by step

guessesTaken	0
myName	Bob

```
# This is a guess the number game.
import random

guessesTaken = 0

print'Hello! What is your name?'
myName = raw_input()

number = random.randint(1, 20)
print'Well, ' + myName + ', I am thinking of a number between 1 and 20.'
```

Code Explanation – step by step

guessesTaken	0
myName	Bob

```
# This is a guess the number game.
import random

guessesTaken = 0

print'Hello! What is your name?'
myName = raw_input()

number = random.randint(1, 20)
print'Well, ' + myName + ', I am thinking of a number between 1 and 20.'

while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1
```

Code Explanation – step by step

guessesTaken	0
myName	Bob
number	8

```
# This is a guess the number game.
import random

guessesTaken = 0

print'Hello! What is your name?'
myName = raw_input()

number = random.randint(1, 20)
print'Well, ' + myName + ', I am thinking of a number between 1 and 20.'

while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1
```

Code Explanation – step by step

guessesTaken	0
myName	Bob
number	8

```
# This is a guess the number game.
import random

guessesTaken = 0

print'Hello! What is your name?'
myName = raw_input()

number = random.randint(1, 20)
print'Well, ' + myName + ', I am thinking of a number between 1 and 20.'

while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1
```

Code Explanation – step by step

guessesTaken	0
myName	Bob
number	8

```
# This is a guess the number game.
import random

guessesTaken = 0

print'Hello! What is your name?'
myName = raw_input()

number = random.randint(1, 20)
print'Well, ' + myName + ', I am thinking of a number between 1 and 20.'

while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1
```

Code Explanation – step by step

```
while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1

    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break
```

guessesTaken	0
myName	Bob
number	8

Code Explanation – step by step

```
while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1

    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break
```

guessesTaken	0
myName	Bob
number	8
guess	‘12’

Code Explanation – step by step

```
while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1

    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break
```

guessesTaken	0
myName	Bob
number	8
guess	12

Code Explanation – step by step

```
while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1

    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break
```

guessesTaken	0
myName	Bob
number	8
guess	12

Code Explanation – step by step

```
while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1

    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break
```

guessesTaken	1
myName	Bob
number	8
guess	12

Code Explanation – step by step

```
while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1

    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break
```

guessesTaken	1
myName	Bob
number	8
guess	12

Code Explanation – step by step

```
while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1

    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break
```

guessesTaken	1
myName	Bob
number	8
guess	12

Code Explanation – step by step

```
while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1

    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break
```

guessesTaken	1
myName	Bob
number	8
guess	12

Code Explanation – step by step

```
while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1

    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break
```

guessesTaken	1
myName	Bob
number	8
guess	12

Code Explanation – step by step

```
while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1

    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break
```

guessesTaken	1
myName	Bob
number	8
guess	12

Code Explanation – step by step

```
while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1

    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break
```

guessesTaken	1
myName	Bob
number	8
guess	'6'

Code Explanation – step by step

```
while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1

    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break
```

guessesTaken	1
myName	Bob
number	8
guess	6

Code Explanation – step by step

```
while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1

    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break
```

guessesTaken	1
myName	Bob
number	8
guess	6

Code Explanation – step by step

```
while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1

    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break
```

guessesTaken	2
myName	Bob
number	8
guess	6

Code Explanation – step by step

```
while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1

    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break
```

guessesTaken	2
myName	Bob
number	8
guess	6

Code Explanation – step by step

```
while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1

    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break
```

guessesTaken	2
myName	Bob
number	8
guess	6

Code Explanation – step by step

```
while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1

    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break
```

guessesTaken	2
myName	Bob
number	8
guess	6

Code Explanation – step by step

```
while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1

    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break
```

guessesTaken	2
myName	Bob
number	8
guess	6

Code Explanation – step by step

```
while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1

    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break
```

guessesTaken	2
myName	Bob
number	8
guess	6

Code Explanation – step by step

```
while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1

    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break
```

guessesTaken	2
myName	Bob
number	8
guess	'8'

Code Explanation – step by step

```
while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1

    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break
```

guessesTaken	2
myName	Bob
number	8
guess	8

Code Explanation – step by step

```
while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1

    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break
```

guessesTaken	2
myName	Bob
number	8
guess	8

Code Explanation – step by step

```
while guessesTaken < 6:
    print 'Take a guess.'
    guess = raw_input()
    guess = int(guess)

    guessesTaken = guessesTaken + 1

    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break
```

guessesTaken	3
myName	Bob
number	8
guess	8

Code Explanation – step by step

```
    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break

if guess == number:
    guessesTaken = str(guessesTaken)
    print 'Good job, ' + myName + '! You guessed my number in '
    + guessesTaken + ' guesses!'

if guess != number:
    number = str(number)
    print 'Nope. The number I was thinking of was ' + number
```

guessesTaken	3
myName	Bob
number	8
guess	8

Code Explanation – step by step

```
    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break

if guess == number:
    guessesTaken = str(guessesTaken)
    print 'Good job, ' + myName + '! You guessed my number in '
    + guessesTaken + ' guesses!'

if guess != number:
    number = str(number)
    print 'Nope. The number I was thinking of was ' + number
```

guessesTaken	3
myName	Bob
number	8
guess	8

Code Explanation – step by step

```
    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break

if guess == number:
    guessesTaken = str(guessesTaken)
    print 'Good job, ' + myName + '! You guessed my number in '
    + guessesTaken + ' guesses!'

if guess != number:
    number = str(number)
    print 'Nope. The number I was thinking of was ' + number
```

guessesTaken	3
myName	Bob
number	8
guess	8

Code Explanation – step by step

```
    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break
```

guessesTaken	3
myName	Bob
number	8
guess	8

```
if guess == number:
    guessesTaken = str(guessesTaken)
    print 'Good job, ' + myName + '! You guessed my number in '
    + guessesTaken + ' guesses!'

if guess != number:
    number = str(number)
    print 'Nope. The number I was thinking of was ' + number
```

Code Explanation – step by step

```
    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break

if guess == number:
    guessesTaken = str(guessesTaken)
    print 'Good job, ' + myName + '! You guessed my number in '
    + guessesTaken + ' guesses!'

if guess != number:
    number = str(number)
    print 'Nope. The number I was thinking of was ' + number
```

guessesTaken	3
myName	Bob
number	8
guess	8

Code Explanation – step by step

```
    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break

if guess == number:
    guessesTaken = str(guessesTaken)
    print 'Good job, ' + myName + '! You guessed my number in '
    + guessesTaken + ' guesses!'

if guess != number:
    number = str(number)
    print 'Nope. The number I was thinking of was ' + number
```

guessesTaken	'3'
myName	Bob
number	8
guess	8

Code Explanation – step by step

```
    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break

if guess == number:
    guessesTaken = str(guessesTaken)
    print 'Good job, ' + myName + '! You guessed my number in '
    + guessesTaken + ' guesses!'

if guess != number:
    number = str(number)
    print 'Nope. The number I was thinking of was ' + number
```

guessesTaken	'3'
myName	Bob
number	8
guess	8

Code Explanation – step by step

```
    if guess < number:
        print 'Your guess is too low.'

    if guess > number:
        print 'Your guess is too high.'

    if guess == number:
        break

if guess == number:
    guessesTaken = str(guessesTaken)
    print 'Good job, ' + myName + '! You guessed my number in '
    + guessesTaken + ' guesses!'

if guess != number:
    number = str(number)
    print 'Nope. The number I was thinking of was ' + number
```

guessesTaken	3
myName	Bob
number	8
guess	8

Some Changes We Could Make

- Try changing this program

```
number = random.randint(1, 20)
print 'Well, ' + myName + ', I am thinking of a number
       between 1 and 20.'
```

```
number = random.randint(1, 100)
print 'Well, ' + myName + ', I am thinking of a number
       between 1 and 20.'
```

Some Changes We Could Make

- Try changing this program

```
while guessesTaken < 6:
```

```
while guessesTaken < 4:
```

What Exactly is Programming?

Programming

- Just the action of **writing codes** for programs (Creating programs).

"But what exactly is a program?"

- **Output**
  - The program decides what exact text to show on the screen.
- **Input**
  - based on its instructions and on the text that the player typed on the keyboard.
- A **program** is a collection of instructions.

What Exactly is Programming?

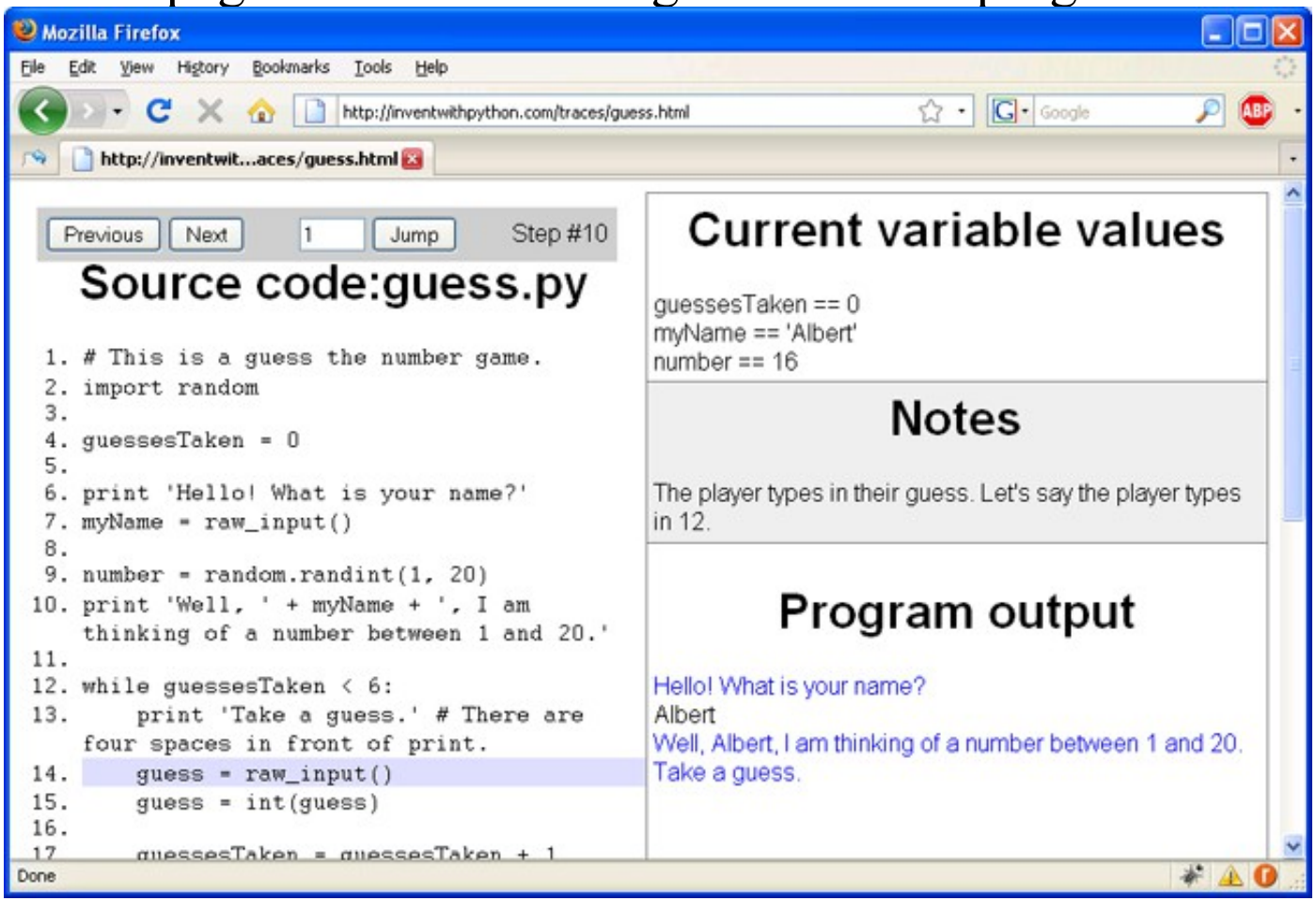
- **"What kind of instructions?"**
  - Expressions
  - Function calls
  - Conditions
  - **flow control** statements
    - if, while and break
  - The print() function, input() function
    - This is called **I/O**

A Web Page for Program Tracing

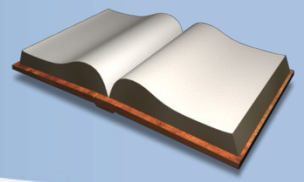
- Go to this book's website at http://inventwithpython.com/traces

A Web Page for Program Tracing

- Find a page that **traces** through each of the programs.

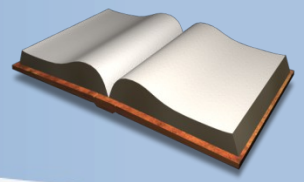


Things Covered In This Chapter(1/2)



- `import` statements
- Modules
- Arguments
- `while` statements
- Conditions
- Blocks
- Comparison operators

Things Covered In This Chapter(2/2)



- The difference between = and ==.
- `if` statements
- The `break` keyword.
- The `str()` function.
- The `random.randint()` function.