ODULE 3 STRINGS

Literals

string values in Python code is fairly straightforward: they begin and end with a
ut then how can you use a quote inside a string?

That is Alice's cat.' won’t work, because Python thinks the string ends after Alice,
t.') is invalid Python code. Fortunately, there are multiple ways to type strings.

Quotes

can begin and end with double quotes, just as they do with single quotes. One be
uble quotes is that the string can have a single quote character in it. Enter the fol
interactive shell:

m = "That is Alice's cat."

e string begins with a double quote, Python knows that the single quote is part
d not marking the end of the string. However, if you need to use both single quo
uotes in the string, you’ll need to use escape characters.

rings

e Characters

pe character consists of a backslash (\) followed by the character you want to add

mple, the escape character for a single quote is \'. You can use this inside a stri
nd ends with single quotes. To see how escape characters work, enter the followi
active shell:

m = 'Say hi to Bob\'s mother.'

e following into the interactive shell:

t("Hello there!\nHow are you?\nI\'m doing fine.")

ere!

you?

g fine.

TRING

rings

place an r before the beginning quotation mark of a string to make it a raw string. mpletely ignores all escape characters and prints any backslash that appears in the nple, enter the following into the interactive shell:

nt(r'That is Carol\'s cat.')

Carol\'s cat.

ne Strings with Triple Quotes

line string in Python begins and ends with either three single quotes or three double Any quotes, tabs, or newlines in between the “triple quotes” are considered part of ython’s indentation rules for blocks do not apply to lines inside a multiline string.

ear Alice,

t has been arrested for catnapping, cat burglary, and extortion.

y,

TRING

g and Slicing Strings

s use indexes and slices the same way lists do. You can think of the string 'H
d!' as a list and each character in the string as an item with a corresponding index.

```
H    e    l    l    o    ,         w    o    r    l    d    !    '
0    1    2    3    4    5    6    7    8    9    10   11   12
```

pace and exclamation point are included in the character count, so 'Hello, world!
cters long, from H at index 0 to ! at index 12.

```
spam = 'Hello, world!'
spam[0]

spam[4]

spam[-1]

spam[0:5]
lo'
spam[:5]
lo'
```

TRING

```
m[7:]
'
```

nd not in Operators with Strings

nd `not in` operators can be used with strings just like with list values. An expression w
ined using `in` or `not in` will evaluate to a Boolean `True` or `False`. Enter the following
e shell:

```
llo' in 'Hello, World'

llo' in 'Hello'

LLO' in 'Hello, World'

in 'spam'

ts' not in 'cats and dogs'
```

TRING

G STRINGS INSIDE OTHER STRINGS

trings inside other strings is a common operation in programming. So far, we've beer
erator and string concatenation to do this:

```
ne = 'Al'
e = 4000
ello, my name is ' + name + '. I am ' + str(age) + ' years old.'
 my name is Al. I am 4000 years old.'
```

OR

```
ne = 'Al'
e = 4000
 name is %s. I am %s years old.' % (name, age)
ne is Al. I am 4000 years old.'
```

3.6 introduced *f-strings*, which is similar to string interpolation except that brac
tead of %s, with the expressions placed directly inside the braces. Like raw stri
ave an f prefix before the starting quotation mark. Enter the following into the inte

TRING

```
ne = 'Al'
 = 4000
y name is {name}. Next year I will be {age + 1}.'
ne is Al. Next year I will be 4001.'
```

tring Methods

per(), lower(), isupper(), and islower() Methods

```
m = 'Hello, world!'
m = spam.upper()
m
, WORLD!'
m = spam.lower()
m
orld!'
```

TRING

```
m = 'Hello, world!'
m.islower()

m.isupper()

LLO'.isupper()

c12345'.islower()

345'.islower()

345'.isupper()
```

TRING

```
ello'.isalpha()

ello123'.isalpha()

ello123'.isalnum()

ello'.isalnum()

23'.isdecimal()

   '.isspace()

his Is Title Case'.istitle()
```

TRING

```
rue:
nt('Enter your age:')
e = input()
 age.isdecimal():
  break
nt('Please enter a number for your age.')
rue:
nt('Select a new password (letters and numbers only):')
sword = input()
 password.isalnum():
  break
nt('Passwords can only have letters and numbers.')
```

TRING

tswith() and endswith() Methods

rtswith() and endswith() methods return True if the string value they are called on (respectively) with the string passed to the method; otherwise, they return False. Er into the interactive shell:

```
ello, world!'.startswith('Hello')

ello, world!'.endswith('world!')

oc123'.startswith('abcdef')

oc123'.endswith('12')

ello, world!'.startswith('Hello, world!')

ello, world!'.endswith('Hello, world!')
```

TRING

ing Whitespace with the strip(), rstrip(), and lstrip() Methods

```
m = '    Hello, World    '
m.strip()
  World'
m.lstrip()
  World    '
m.rstrip()
ello, World'
```

RIC VALUES OF CHARACTERS WITH THE ORD() AND CHR() FUNCTIONS

```
('A')

('4')

('!')

(65)
```

TRING