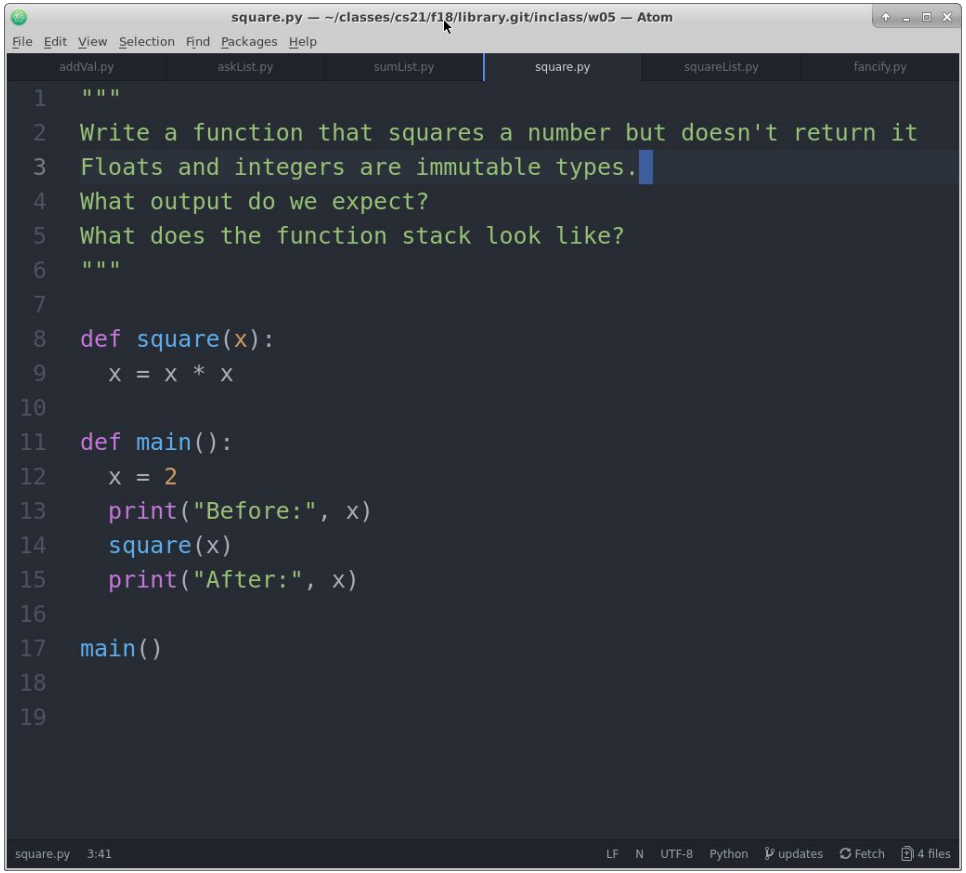
Changing values of parameters

If a variable is **immutable** and you change its value inside a function, the change **does not last** past the lifetime of the function

If a variable is **mutable** and you change its value inside a function, the change is permanent

Example: square.py

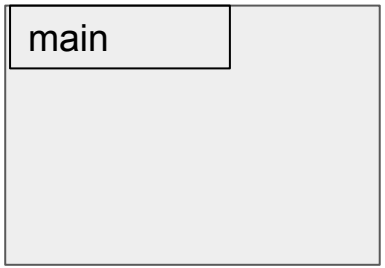
```
almond[w05]$ python3 square.py

Before: 2

After: 2
```

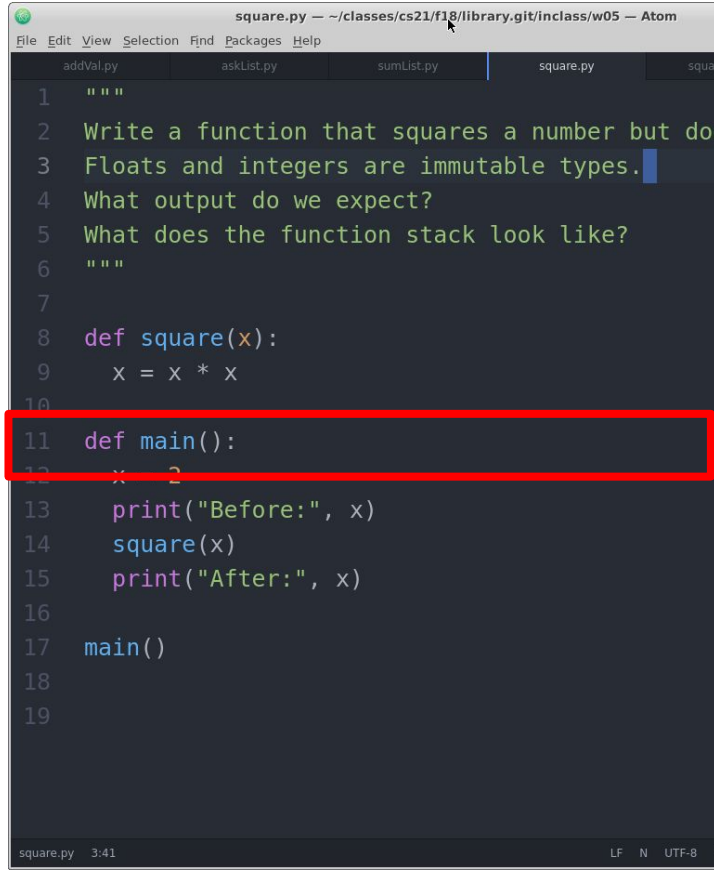


Function stack diagram

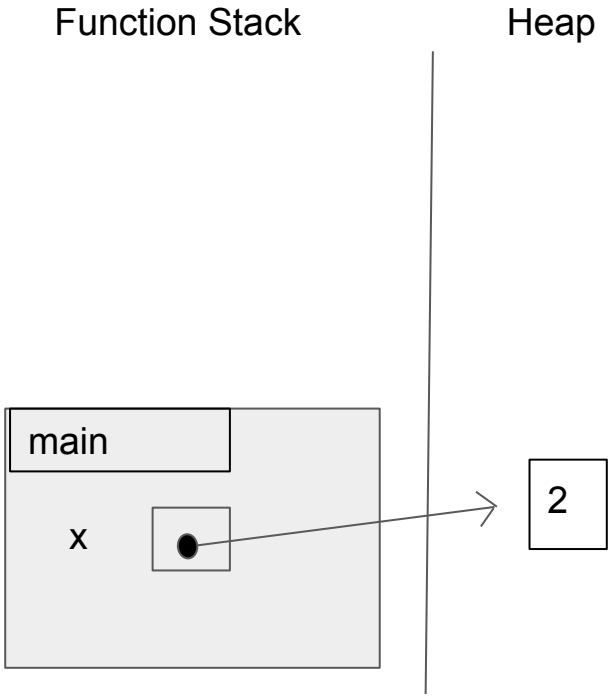
Function Stack

Heap





Function stack diagram



square.py — ~/classes/cs21/f18/library.git/inclass/w05 — Atom

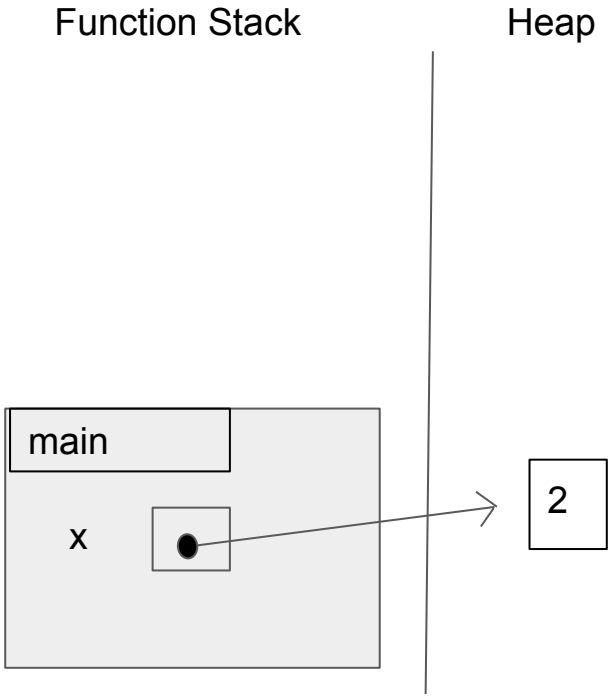
File Edit View Selection Find Packages Help

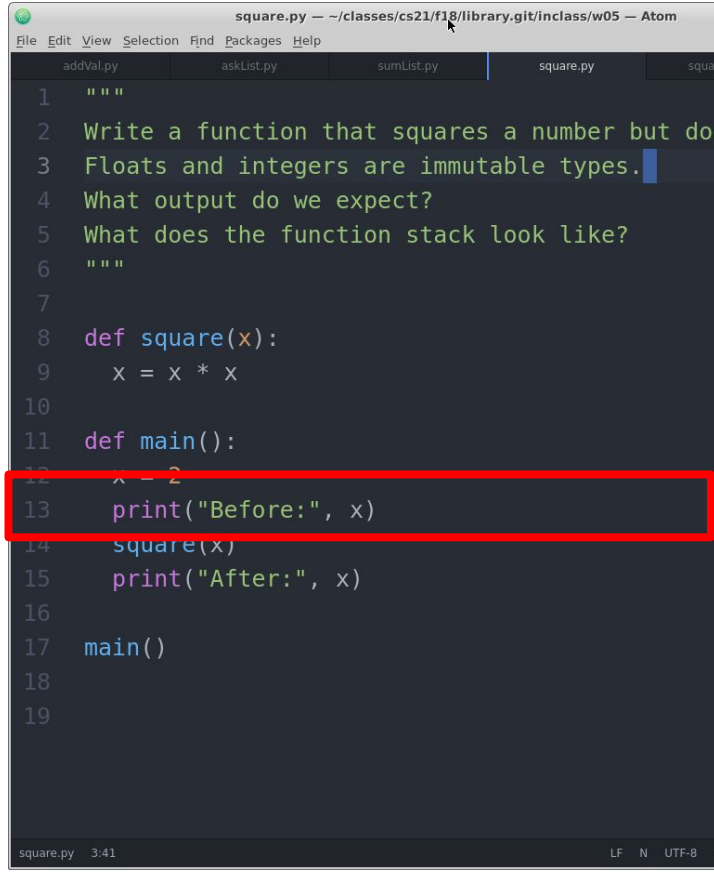
addVal.py | askList.py | sumList.py | square.py

```
1  """
2  Write a function that squares a number but do
3  Floats and integers are immutable types.
4  What output do we expect?
5  What does the function stack look like?
6  """
7
8  def square(x):
9    x = x * x
10
11 def main():
12   x = 2
13   print("Before:", x)
14   square(x)
15   print("After:", x)
16
17 main()
18
19
```

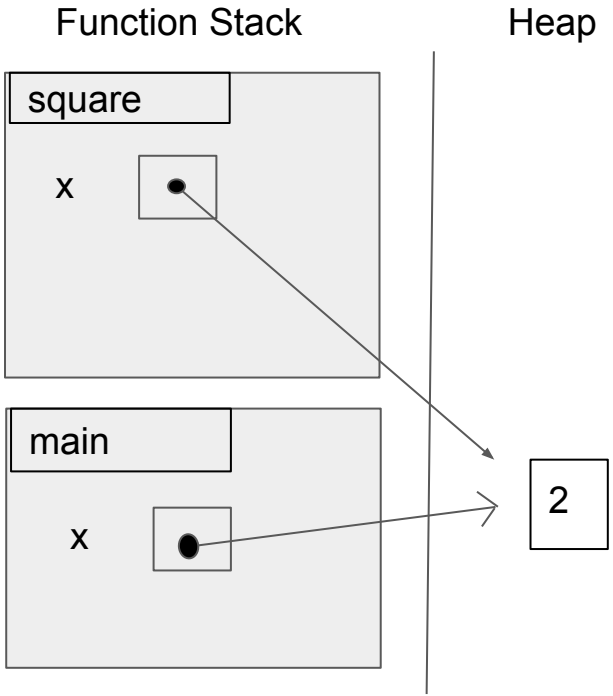
square.py 3:41 LF N UTF-8

Function stack diagram





Function stack diagram



square.py — ~/classes/cs21/f18/library.git/inclass/w05 — Atom

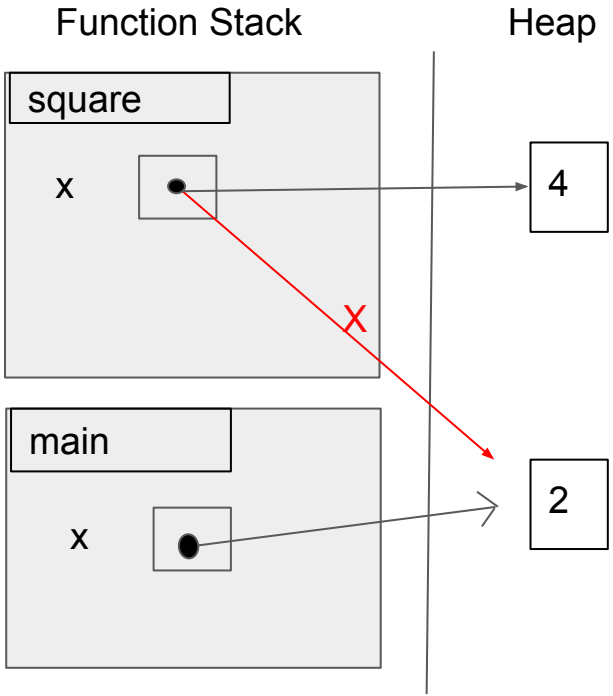
File Edit View Selection Find Packages Help

addVal.py | askList.py | sumList.py | square.py

```
1  """
2  Write a function that squares a number but do
3  Floats and integers are immutable types.
4  What output do we expect?
5  What does the function stack look like?
6  """
7
8  def square(x):
9    x = x * x
10
11 def main():
12   x = 2
13   print("Before:", x)
14   square(x)
15   print("After:", x)
16
17 main()
18
19
```

square.py 3:41 LF N UTF-8

Function stack diagram



square.py — ~/classes/cs21/f18/library.git/inclass/w05 — Atom

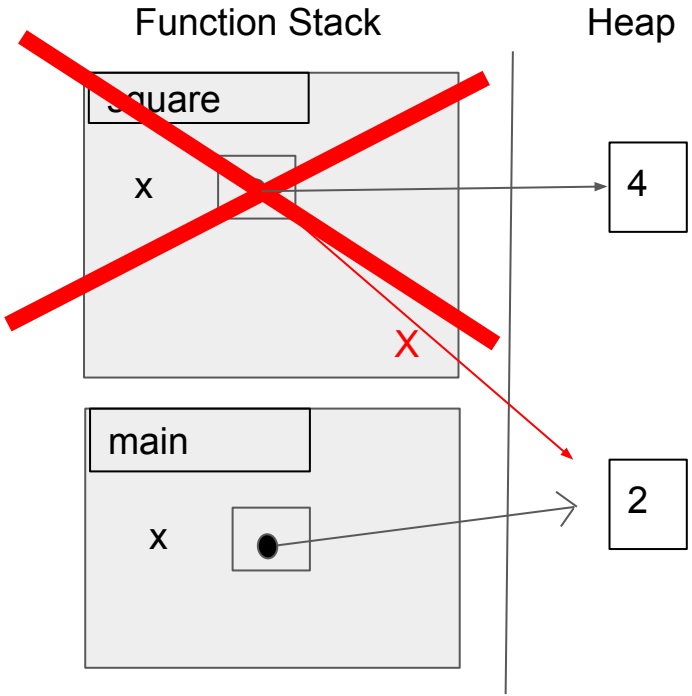
File Edit View Selection Find Packages Help

addVal.py | askList.py | sumList.py | square.py

```
1  """
2  Write a function that squares a number but do
3  Floats and integers are immutable types.
4  What output do we expect?
5  What does the function stack look like?
6  """
7
8  def square(x):
9    x = x * x
10
11 def main():
12   x = 2
13   print("Before:", x)
14   square(x)
15   print("After:", x)
16
17 main()
18
19
```

square.py 3:41 LF N UTF-8

Function stack diagram



square.py — ~/classes/cs21/f18/library.git/inclass/w05 — Atom

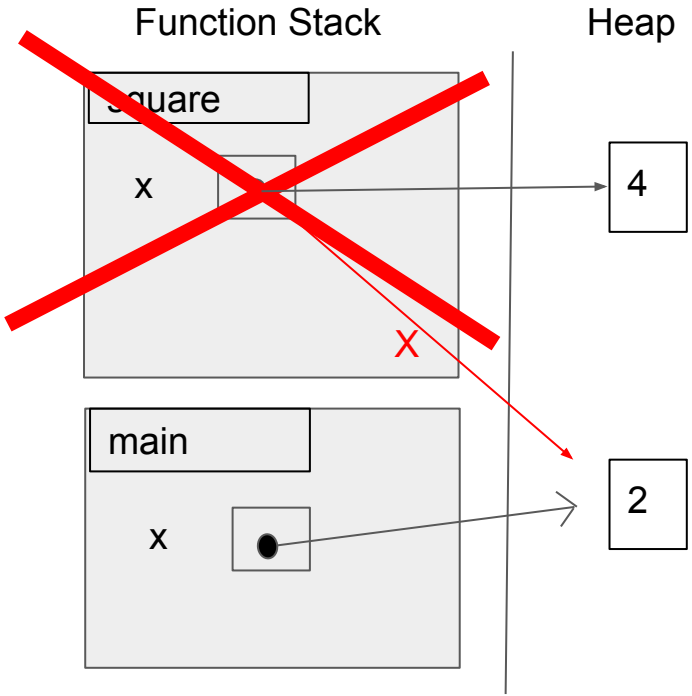
File Edit View Selection Find Packages Help

addVal.py | askList.py | sumList.py | square.py

```
"""
Write a function that squares a number but do
Floats and integers are immutable types.
What output do we expect?
What does the function stack look like?
"""

def square(x):
  x = x * x

def main():
  x = 2
  print("Before:", x)
  square(x)
  print("After:", x)

main()
```

square.py 3:41 LF N UTF-8

Function stack diagram



square.py — ~/classes/cs21/f18/library.git/inclass/w05 — Atom

File Edit View Selection Find Packages Help

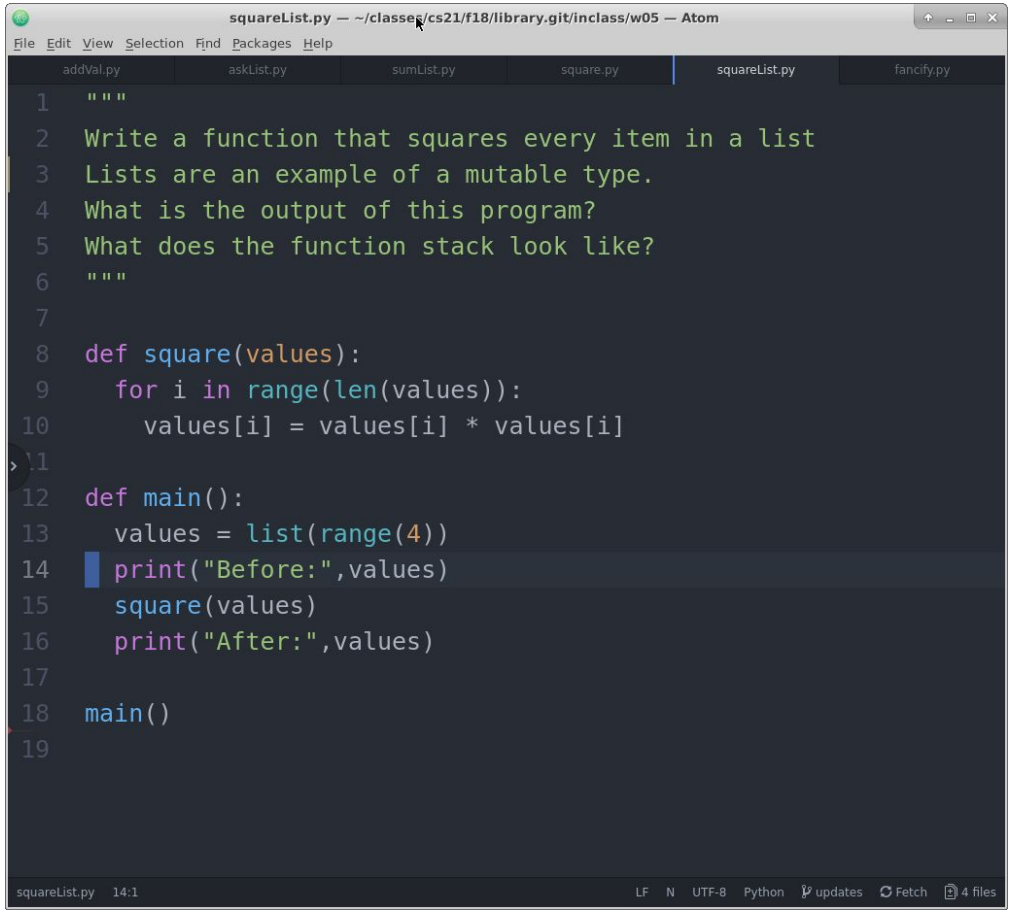
addVal.py | askList.py | sumList.py | square.py

```
1  """
2  Write a function that squares a number but do
3  Floats and integers are immutable types.
4  What output do we expect?
5  What does the function stack look like?
6  """
7
8  def square(x):
9    x = x * x
10
11 def main():
12   x = 2
13   print("Before:", x)
14   square(x)
15   print("After:", x)
16
17 main()
18
19
```

square.py 3:41 LF N UTF-8

Example: squareList.py

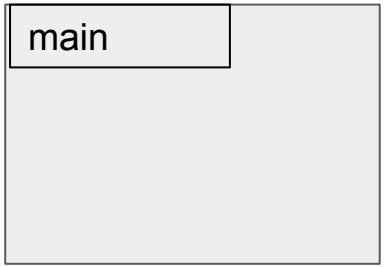
```
almond[w05]$ python3 squareList.py

Before: [0, 1, 2, 3]

After: [0, 1, 4, 9]
```



Function Stack diagram

Function Stack

Heap



squareList.py — ~/classes/cs21/f18/library.git/inclass/w05 — Atom

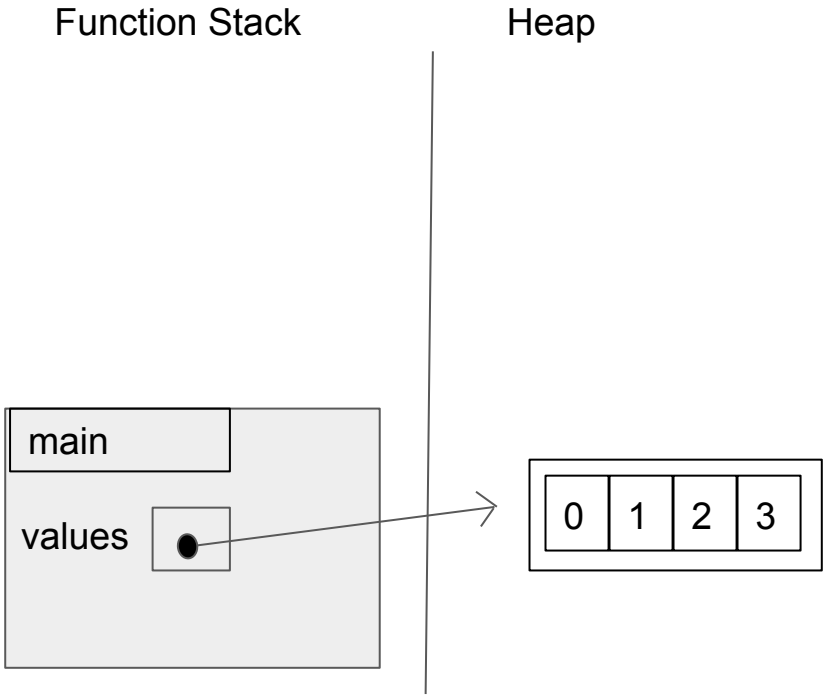
File Edit View Selection Find Packages Help

addVal.py askList.py sumList.py square.py squareList.p

```
"""
Write a function that squares every item in a li
Lists are an example of a mutable type.
What is the output of this program?
What does the function stack look like?
"""

def square(values):
  for i in range(len(values)):
    values[i] = values[i] * values[i]

def main():
  values = list(range(4))
  print("Before:",values)
  square(values)
  print("After:",values)

main()
```

squareList.py 14:1 LF N UTF-8 Python

Function Stack diagram



squareList.py — ~/classes/cs21/f18/library.git/inclass/w05 — Atom

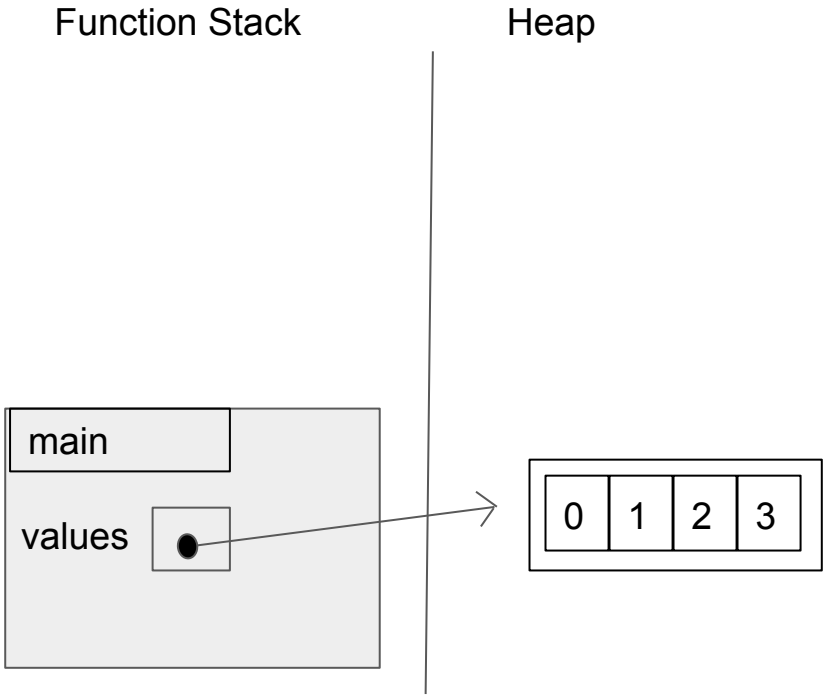
File Edit View Selection Find Packages Help

addVal.py | askList.py | sumList.py | square.py | squareList.p

```
1  """
2  Write a function that squares every item in a li
3  Lists are an example of a mutable type.
4  What is the output of this program?
5  What does the function stack look like?
6  """
7
8  def square(values):
9    for i in range(len(values)):
10     values[i] = values[i] * values[i]
11
12 def main():
13   values = list(range(4))
14   print("Before:",values)
15   square(values)
16   print("After:",values)
17
18 main()
19
```

squareList.py 14:1 LF N UTF-8 Python

Function Stack diagram



squareList.py — ~/classes/cs21/f18/library.git/inclass/w05 — Atom

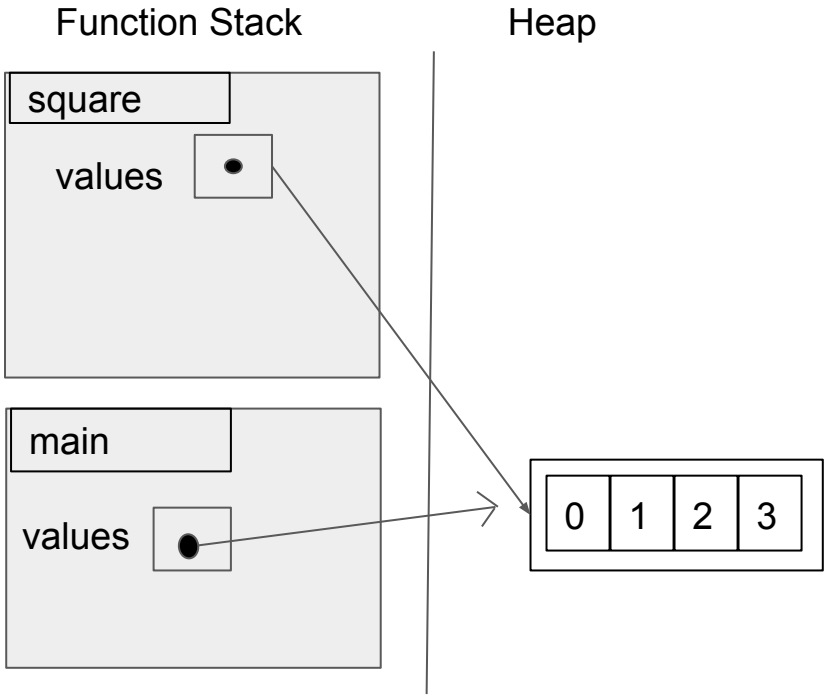
File Edit View Selection Find Packages Help

addVal.py | askList.py | sumList.py | square.py | squareList.p

```
"""
Write a function that squares every item in a li
Lists are an example of a mutable type.
What is the output of this program?
What does the function stack look like?
"""

def square(values):
  for i in range(len(values)):
    values[i] = values[i] * values[i]

def main():
  values = list(range(4))
  print("Before:",values)
  square(values)
  print("After:",values)

main()
```

squareList.py 14:1 LF N UTF-8 Python

Function Stack diagram



squareList.py — ~/classes/cs21/f18/library.git/inclass/w05 — Atom

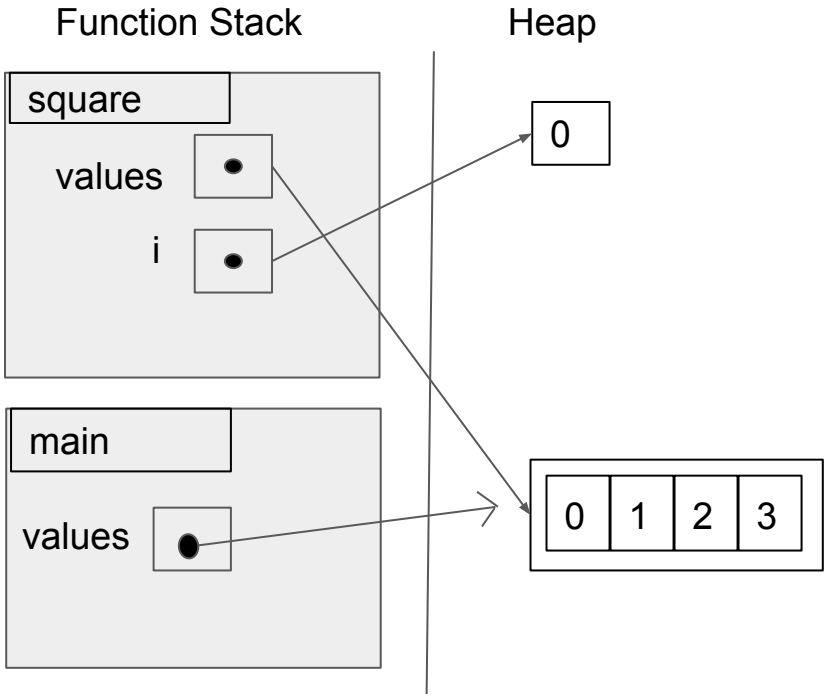
File Edit View Selection Find Packages Help

addVal.py askList.py sumList.py square.py squareList.p

```
1  """
2  Write a function that squares every item in a li
3  Lists are an example of a mutable type.
4  What is the output of this program?
5  What does the function stack look like?
6  """
7
8  def square(values):
9    for i in range(len(values)):
10     values[i] = values[i] * values[i]
11
12 def main():
13   values = list(range(4))
14   print("Before:",values)
15   square(values)
16   print("After:",values)
17
18 main()
19
```

squareList.py 14:1 LF N UTF-8 Python

Function Stack diagram



squareList.py — ~/classes/cs21/f18/library.git/inclass/w05 — Atom

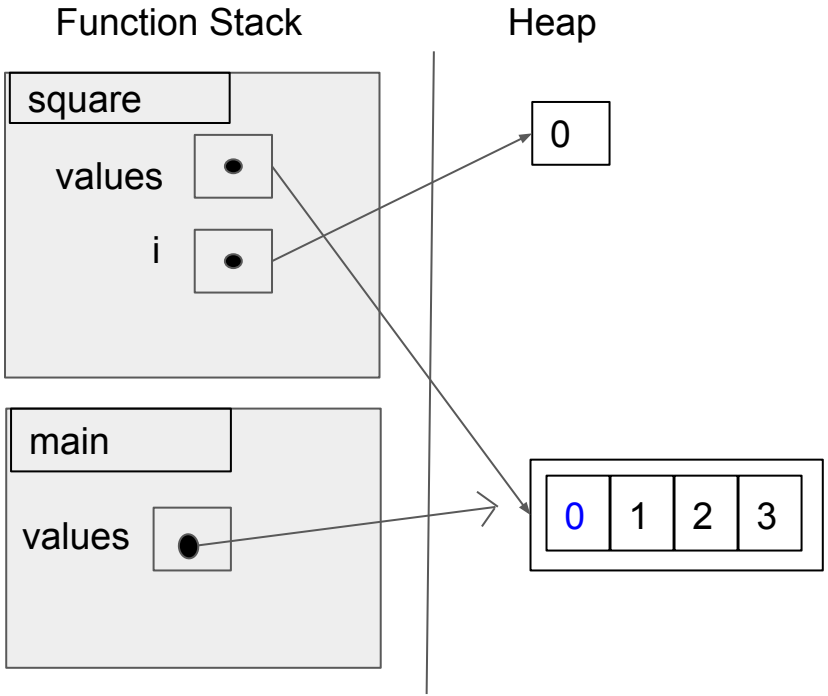
File Edit View Selection Find Packages Help

addVal.py askList.py sumList.py square.py squareList.p

```
1  """
2  Write a function that squares every item in a li
3  Lists are an example of a mutable type.
4  What is the output of this program?
5  What does the function stack look like?
6  """
7
8  def square(values):
9      for i in range(len(values)):
10         values[i] = values[i] * values[i]
11
12 def main():
13     values = list(range(4))
14     print("Before:",values)
15     square(values)
16     print("After:",values)
17
18 main()
19
```

squareList.py 14:1 LF N UTF-8 Python

Function Stack diagram



squareList.py — ~/classes/cs21/f18/library.git/inclass/w05 — Atom

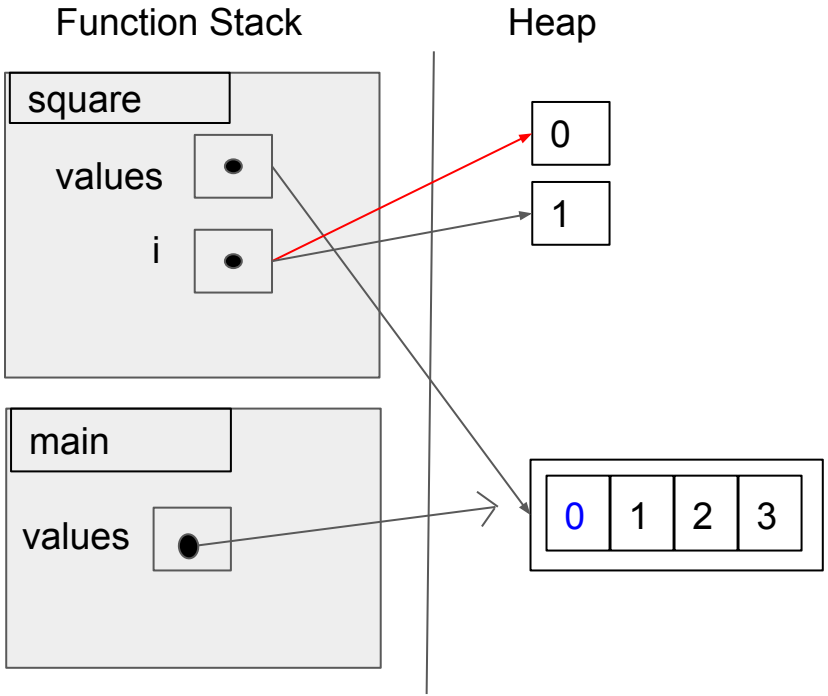
File Edit View Selection Find Packages Help

addVal.py askList.py sumList.py square.py squareList.p

```
1  """
2  Write a function that squares every item in a li
3  Lists are an example of a mutable type.
4  What is the output of this program?
5  What does the function stack look like?
6  """
7
8  def square(values):
9      for i in range(len(values)):
10         values[i] = values[i] * values[i]
11
12 def main():
13     values = list(range(4))
14     print("Before:",values)
15     square(values)
16     print("After:",values)
17
18 main()
19
```

squareList.py 14:1 LF N UTF-8 Python

Function Stack diagram



squareList.py — ~/classes/cs21/f18/library.git/inclass/w05 — Atom

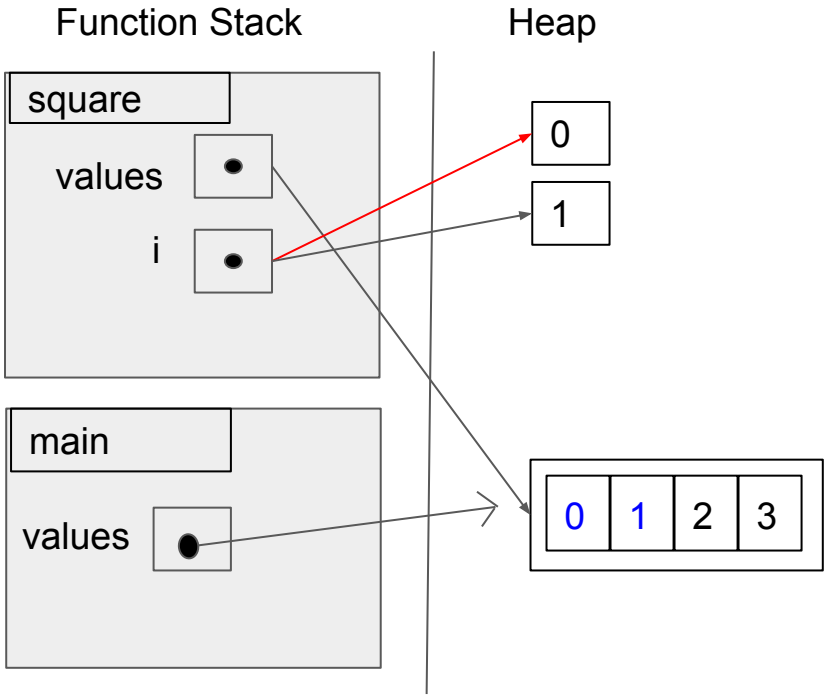
File Edit View Selection Find Packages Help

addVal.py askList.py sumList.py square.py squareList.p

```
"""
Write a function that squares every item in a li
Lists are an example of a mutable type.
What is the output of this program?
What does the function stack look like?
"""

def square(values):
    for i in range(len(values)):
        values[i] = values[i] * values[i]

def main():
    values = list(range(4))
    print("Before:",values)
    square(values)
    print("After:",values)

main()
```

squareList.py 14:1 LF N UTF-8 Python

Function Stack diagram



squareList.py — ~/classes/cs21/f18/library.git/inclass/w05 — Atom

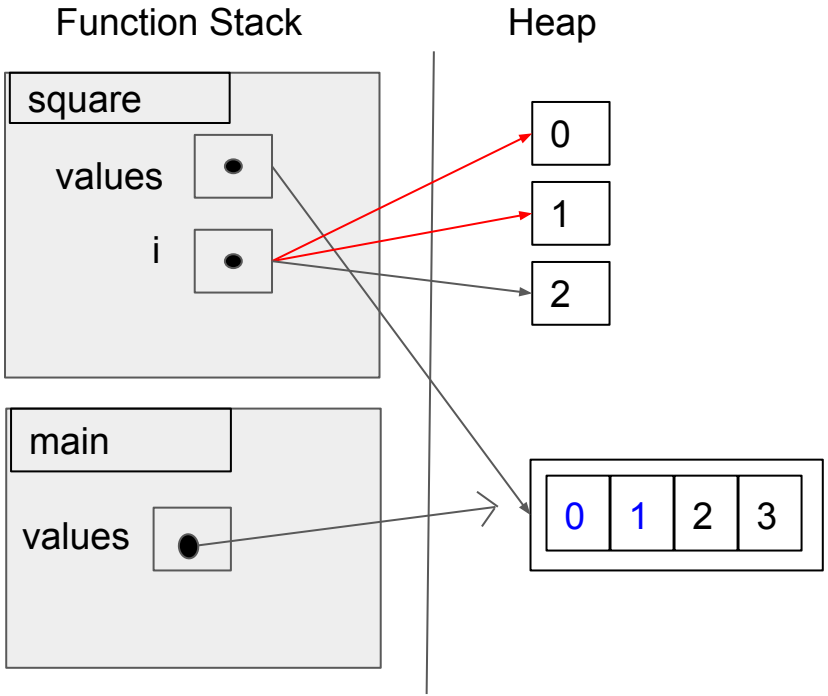
File Edit View Selection Find Packages Help

addVal.py askList.py sumList.py square.py squareList.p

```
1  """
2  Write a function that squares every item in a li
3  Lists are an example of a mutable type.
4  What is the output of this program?
5  What does the function stack look like?
6  """
7
8  def square(values):
9      for i in range(len(values)):
10         values[i] = values[i] * values[i]
11
12 def main():
13     values = list(range(4))
14     print("Before:",values)
15     square(values)
16     print("After:",values)
17
18 main()
19
```

squareList.py 14:1 LF N UTF-8 Python

Function Stack diagram



squareList.py — ~/classes/cs21/f18/library.git/inclass/w05 — Atom

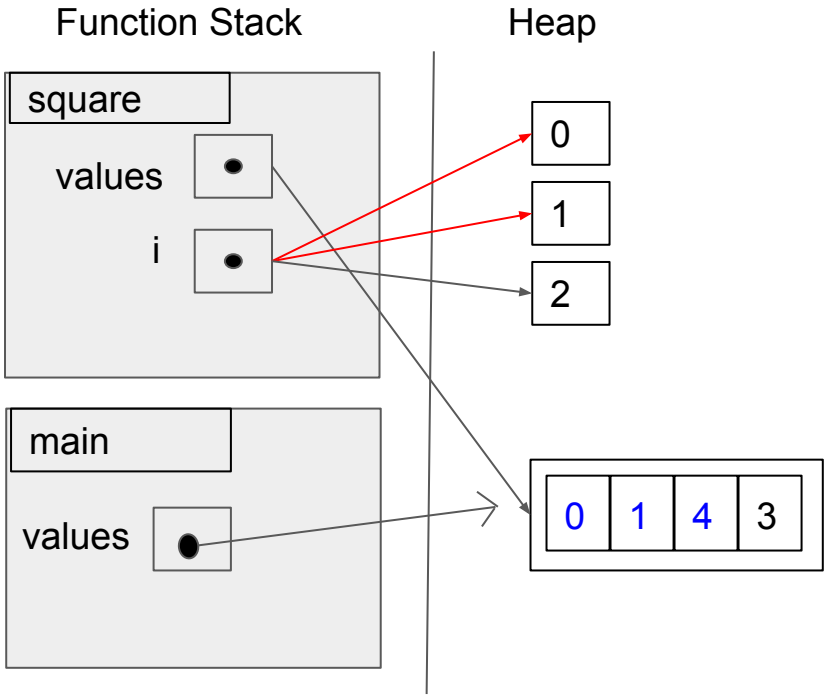
File Edit View Selection Find Packages Help

addVal.py askList.py sumList.py square.py squareList.p

```
1  """
2  Write a function that squares every item in a li
3  Lists are an example of a mutable type.
4  What is the output of this program?
5  What does the function stack look like?
6  """
7
8  def square(values):
9      for i in range(len(values)):
10         values[i] = values[i] * values[i]
11
12 def main():
13     values = list(range(4))
14     print("Before:",values)
15     square(values)
16     print("After:",values)
17
18 main()
19
```

squareList.py 14:1 LF N UTF-8 Python

Function Stack diagram



squareList.py — ~/classes/cs21/f18/library.git/inclass/w05 — Atom

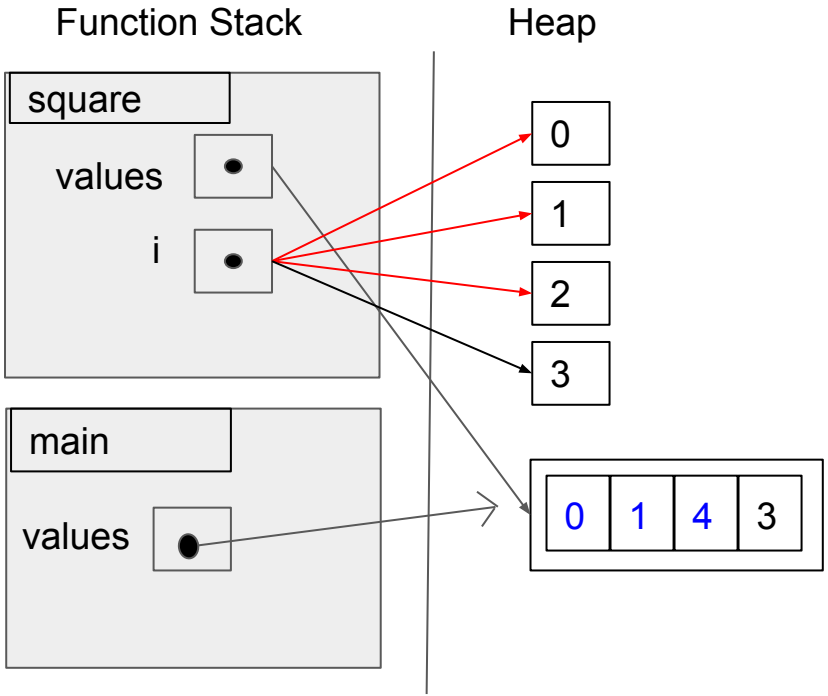
File Edit View Selection Find Packages Help

addVal.py askList.py sumList.py square.py squareList.p

```
"""
Write a function that squares every item in a li
Lists are an example of a mutable type.
What is the output of this program?
What does the function stack look like?
"""

def square(values):
    for i in range(len(values)):
        values[i] = values[i] * values[i]

def main():
    values = list(range(4))
    print("Before:",values)
    square(values)
    print("After:",values)

main()
```

squareList.py 14:1 LF N UTF-8 Python

Function Stack diagram



squareList.py — ~/classes/cs21/f18/library.git/inclass/w05 — Atom

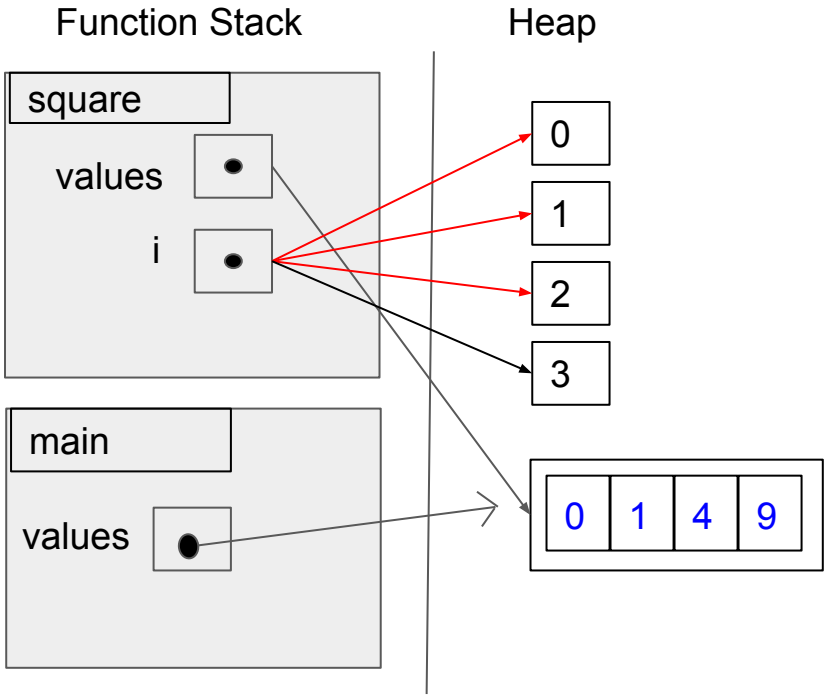
File Edit View Selection Find Packages Help

addVal.py askList.py sumList.py square.py squareList.p

```
1  """
2  Write a function that squares every item in a li
3  Lists are an example of a mutable type.
4  What is the output of this program?
5  What does the function stack look like?
6  """
7
8  def square(values):
9      for i in range(len(values)):
10         values[i] = values[i] * values[i]
11
12 def main():
13     values = list(range(4))
14     print("Before:",values)
15     square(values)
16     print("After:",values)
17
18 main()
19
```

squareList.py 14:1 LF N UTF-8 Python

Function Stack diagram



squareList.py — ~/classes/cs21/f18/library.git/inclass/w05 — Atom

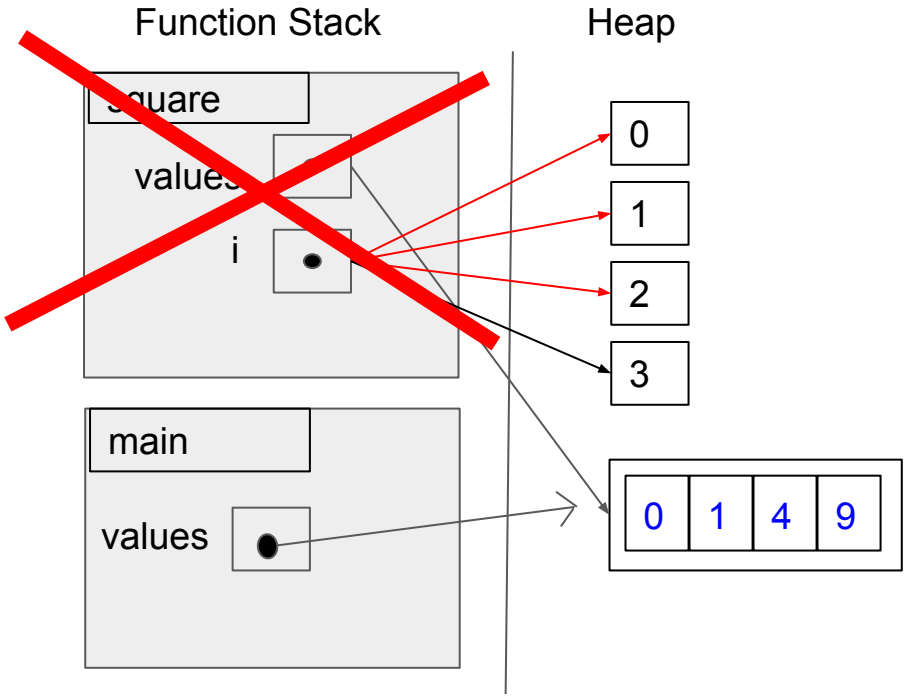
File Edit View Selection Find Packages Help

addVal.py askList.py sumList.py square.py squareList.p

```
1  """
2  Write a function that squares every item in a li
3  Lists are an example of a mutable type.
4  What is the output of this program?
5  What does the function stack look like?
6  """
7
8  def square(values):
9      for i in range(len(values)):
10         values[i] = values[i] * values[i]
11
12 def main():
13     values = list(range(4))
14     print("Before:",values)
15     square(values)
16     print("After:",values)
17
18 main()
19
```

squareList.py 14:1 LF N UTF-8 Python

Function Stack diagram



squareList.py — ~/classes/cs21/f18/library.git/inclass/w05 — Atom

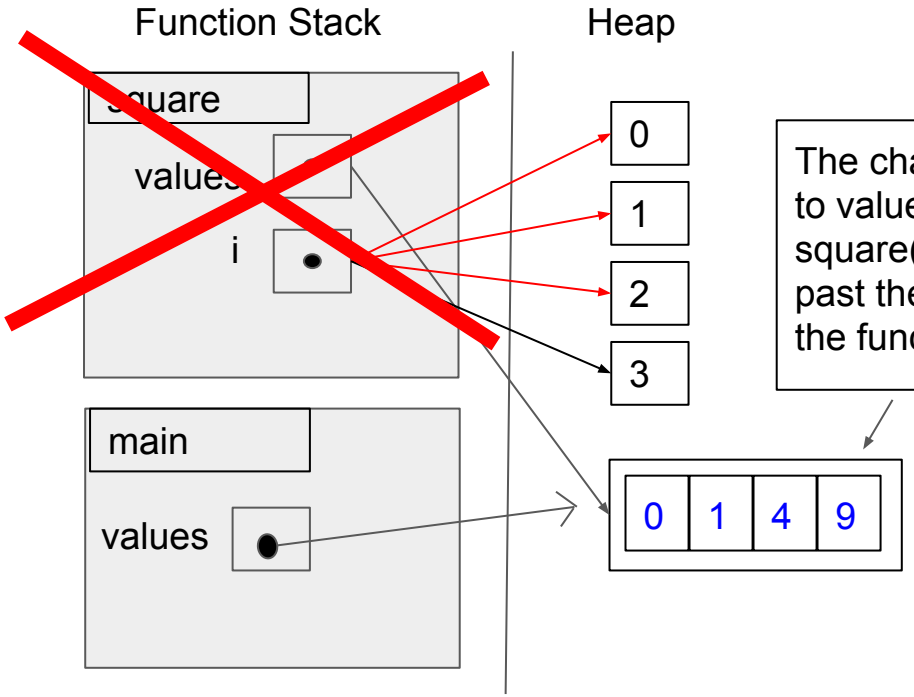
File Edit View Selection Find Packages Help

addVal.py askList.py sumList.py square.py squareList.p

```
"""
Write a function that squares every item in a li
Lists are an example of a mutable type.
What is the output of this program?
What does the function stack look like?
"""

def square(values):
  for i in range(len(values)):
    values[i] = values[i] * values[i]

def main():
  values = list(range(4))
  print("Before:",values)
  square(values)
  print("After:",values)

main()
```

squareList.py 14:1 LF N UTF-8 Python

Function Stack diagram



squareList.py — ~/classes/cs21/f18/library.git/inclass/w05 — Atom

File Edit View Selection Find Packages Help

addVal.py | askList.py | sumList.py | square.py | squareList.p

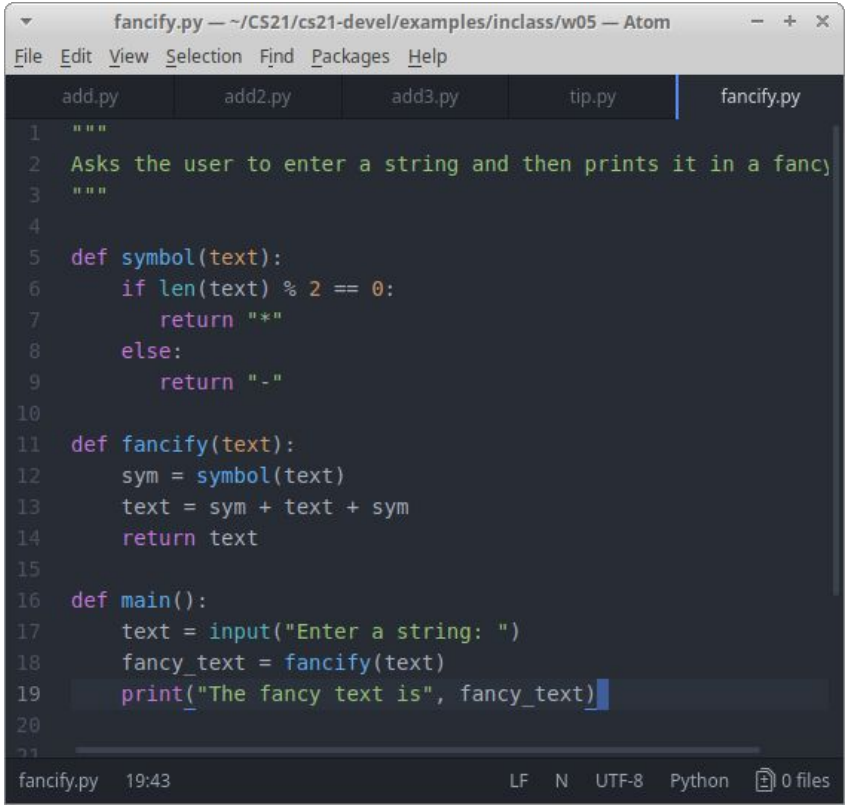
```
1  """
2  Write a function that squares every item in a li
3  Lists are an example of a mutable type.
4  What is the output of this program?
5  What does the function stack look like?
6  """

   def square(values):
     for i in range(len(values)):
       values[i] = values[i] * values[i]

   def main():
     values = list(range(4))
     print("Before:",values)
     square(values)
     print("After:",values)

   main()
```

squareList.py 14:1 LF N UTF-8 Python

fancify.py

strings are immutable types, so they work on the heap similarly to ints, floats, and booleans

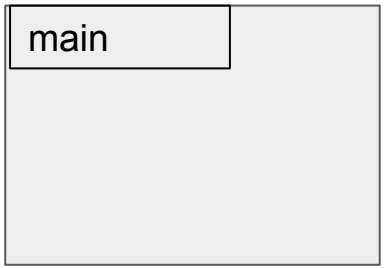
```
almond[w05]$ python3 fancify.py
Enter a string: hello
The fancy text is -hello-

almond[w05]$ python3 fancify.py
Enter a string: cats
The fancy text is *cats*
```



Function stack diagram

Function Stack

Heap



fancify.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom

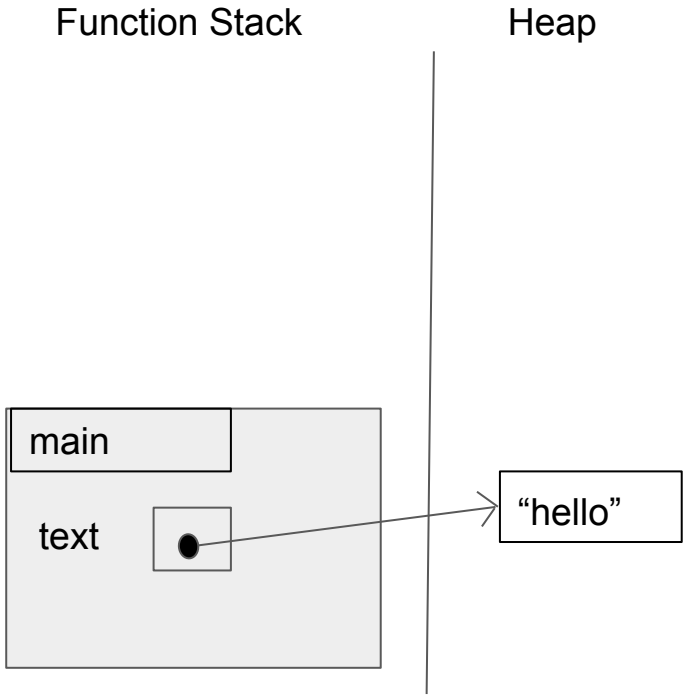
File Edit View Selection Find Packages Help

add.py | add2.py | add3.py | tip.py | fancify.py

```
"""
Asks the user to enter a string and then prints it in a fancy
"""

def symbol(text):
    if len(text) % 2 == 0:
        return "*"
    else:
        return "-"

def fancify(text):
    sym = symbol(text)
    text = sym + text + sym
    return text

def main():
    text = input("Enter a string: ")
    fancy_text = fancify(text)
    print("The fancy text is", fancy_text)
```

fancify.py 19:43 LF N UTF-8 Python 0 files

Function stack diagram



fancify.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom

File Edit View Selection Find Packages Help

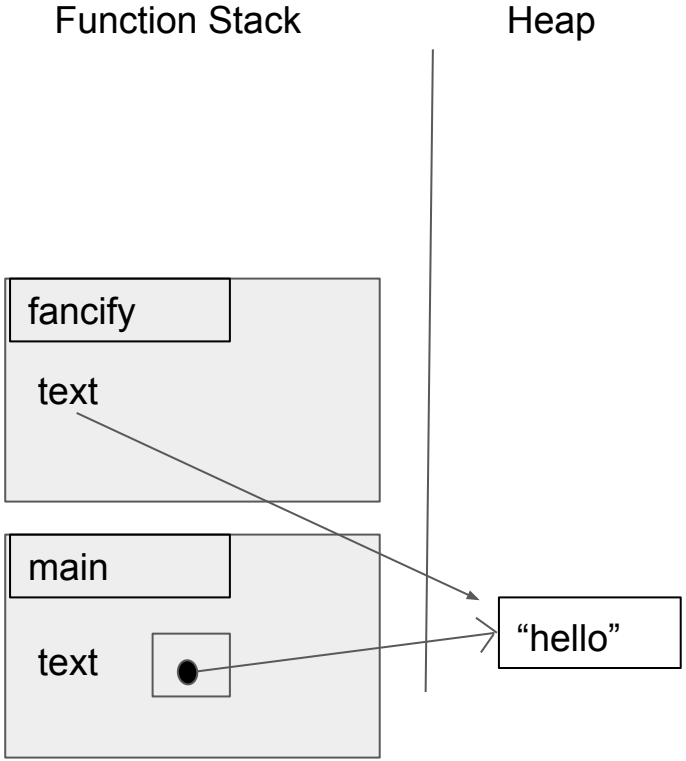
add.py | add2.py | add3.py | tip.py | fancify.py

```
"""
Asks the user to enter a string and then prints it in a fancy
"""

def symbol(text):
    if len(text) % 2 == 0:
        return "*"
    else:
        return "-"

def fancify(text):
    sym = symbol(text)
    text = sym + text + sym
    return text

def main():
    text = input("Enter a string: ")
    fancy_text = fancify(text)
    print("The fancy text is", fancy_text)
```

Assume user enters “hello”

fancify.py 19:43 LF N UTF-8 Python 0 files

Function stack diagram



fancify.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom

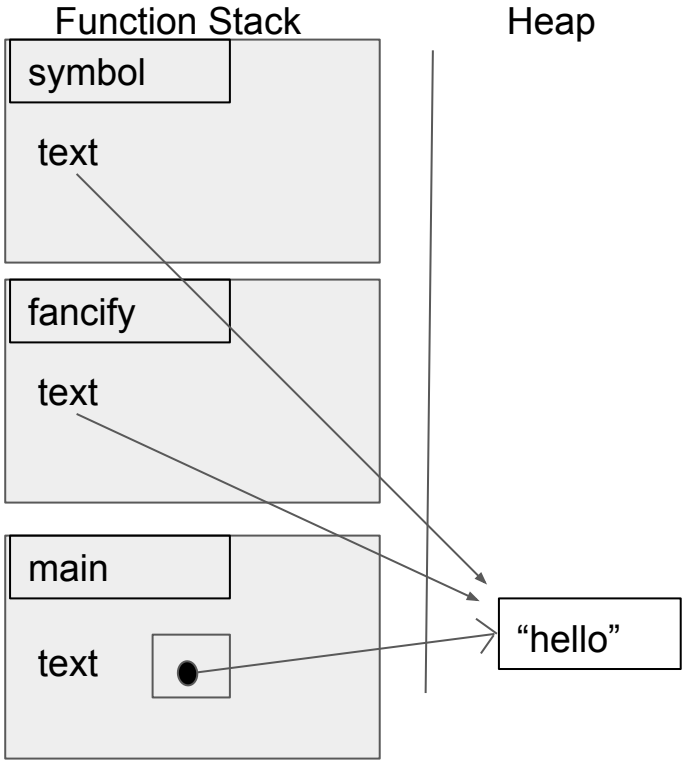
File Edit View Selection Find Packages Help

add.py | add2.py | add3.py | tip.py | fancify.py

```
"""
Asks the user to enter a string and then prints it in a fancy
"""

def symbol(text):
    if len(text) % 2 == 0:
        return "*"
    else:
        return "-"

def fancify(text):
    sym = symbol(text)
    text = sym + text + sym
    return text

def main():
    text = input("Enter a string: ")
    fancy_text = fancify(text)
    print("The fancy text is", fancy_text)
```

fancify.py 19:43 LF N UTF-8 Python 0 files

Function stack diagram



fancify.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom

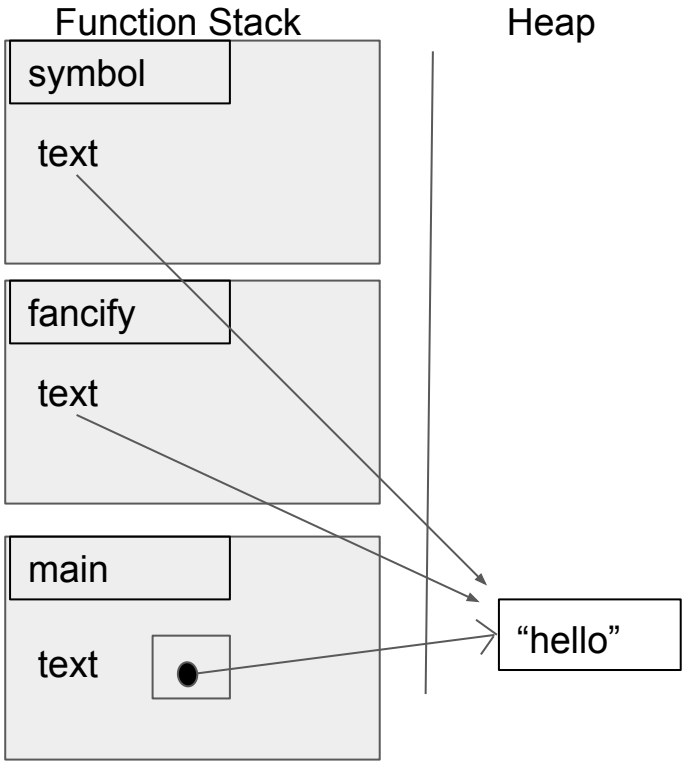
File Edit View Selection Find Packages Help

add.py | add2.py | add3.py | tip.py | fancify.py

```
"""
Asks the user to enter a string and then prints it in a fancy
"""

def symbol(text):
    if len(text) % 2 == 0:
        return "*"
    else:
        return "-"

def fancify(text):
    sym = symbol(text)
    text = sym + text + sym
    return text

def main():
    text = input("Enter a string: ")
    fancy_text = fancify(text)
    print("The fancy text is", fancy_text)
```

fancify.py 19:43 LF N UTF-8 Python 0 files

Function stack diagram



fancify.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom

File Edit View Selection Find Packages Help

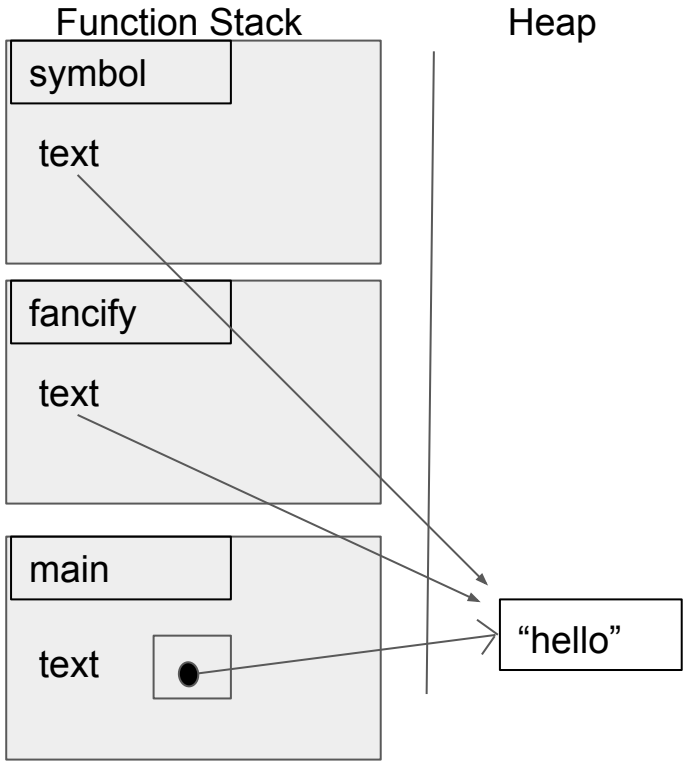
add.py | add2.py | add3.py | tip.py | fancify.py

```
"""
Asks the user to enter a string and then prints it in a fancy
"""

def symbol(text):
    if len(text) % 2 == 0:
        return "*"
    else:
        return "-"

def fancify(text):
    sym = symbol(text)
    text = sym + text + sym
    return text

def main():
    text = input("Enter a string: ")
    fancy_text = fancify(text)
    print("The fancy text is", fancy_text)
```

condition is False

fancify.py 19:43 LF N UTF-8 Python 0 files

Function stack diagram



fancify.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom

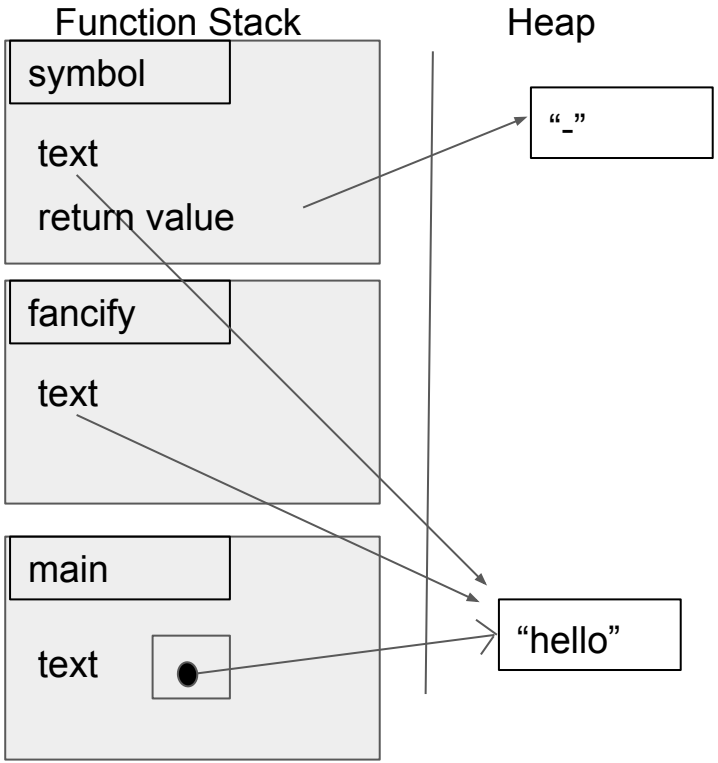
File Edit View Selection Find Packages Help

add.py | add2.py | add3.py | tip.py | fancify.py

```
"""
Asks the user to enter a string and then prints it in a fancy
"""

def symbol(text):
    if len(text) % 2 == 0:
        return "*"
    else:
        return "."

def fancify(text):
    sym = symbol(text)
    text = sym + text + sym
    return text

def main():
    text = input("Enter a string: ")
    fancy_text = fancify(text)
    print("The fancy text is", fancy_text)
```

fancify.py 19:43 LF N UTF-8 Python 0 files

Function stack diagram



fancify.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom

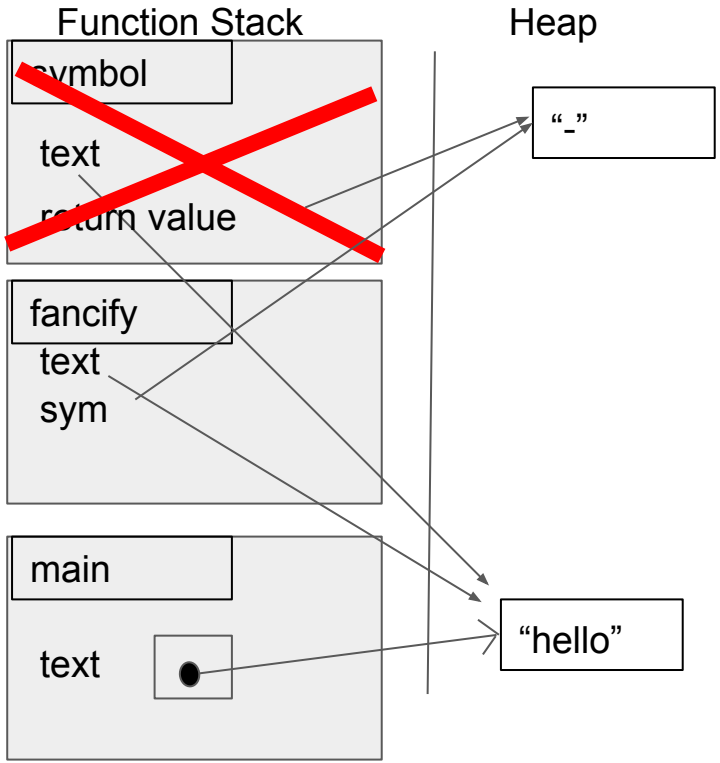
File Edit View Selection Find Packages Help

add.py | add2.py | add3.py | tip.py | fancify.py

```
"""
Asks the user to enter a string and then prints it in a fancy
"""

def symbol(text):
    if len(text) % 2 == 0:
        return "*"
    else:
        return "-"

def fancify(text):
    sym = symbol(text)
    text = sym + text + sym
    return text

def main():
    text = input("Enter a string: ")
    fancy_text = fancify(text)
    print("The fancy text is", fancy_text)
```

fancify.py 19:43 LF N UTF-8 Python 0 files

Function stack diagram



fancify.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom

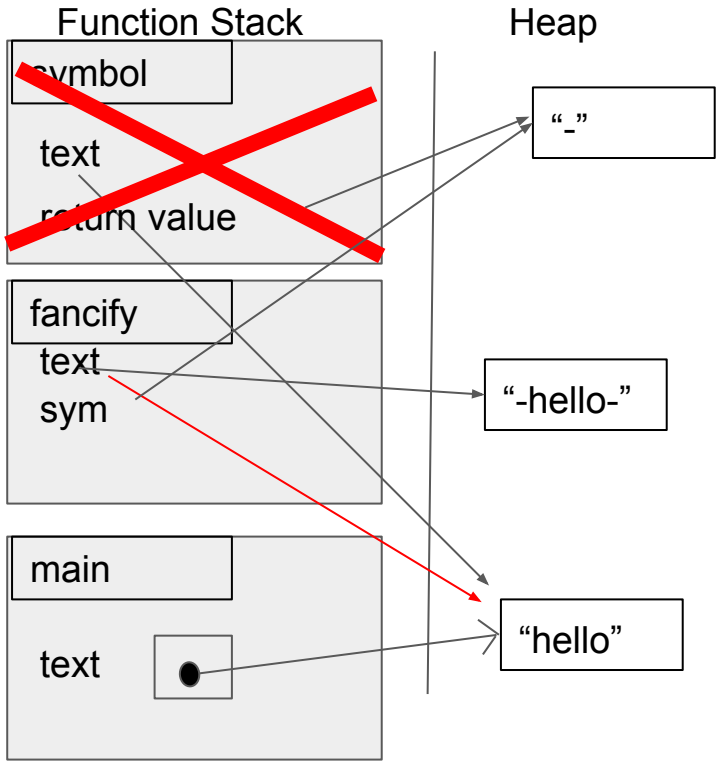
File Edit View Selection Find Packages Help

add.py | add2.py | add3.py | tip.py | fancify.py

```
"""
Asks the user to enter a string and then prints it in a fancy
"""

def symbol(text):
    if len(text) % 2 == 0:
        return "*"
    else:
        return "-"

def fancify(text):
    sym = symbol(text)
    text = sym + text + sym
    return text

def main():
    text = input("Enter a string: ")
    fancy_text = fancify(text)
    print("The fancy text is", fancy_text)
```

fancify.py 19:43 LF N UTF-8 Python 0 files

Function stack diagram



fancify.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom

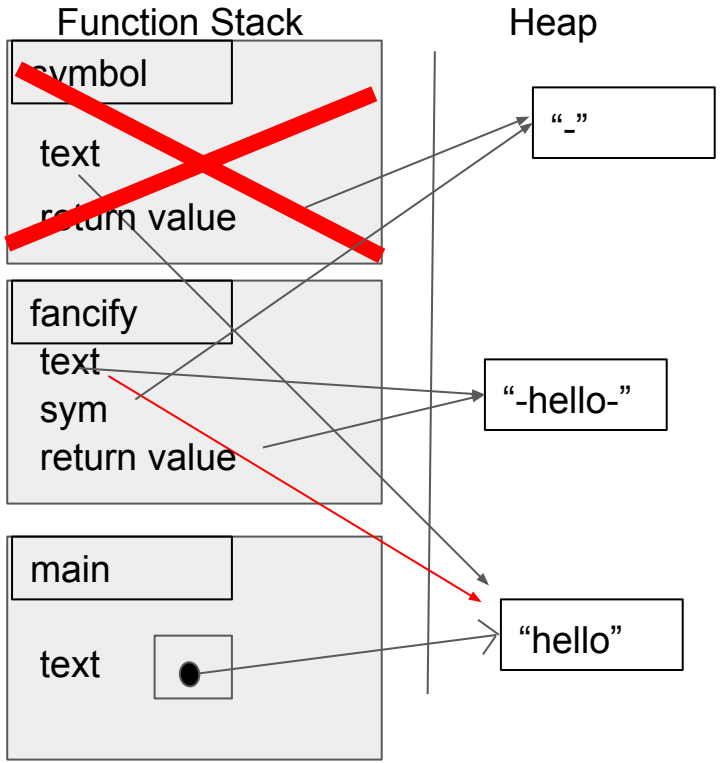
File Edit View Selection Find Packages Help

add.py | add2.py | add3.py | tip.py | fancify.py

```
"""
Asks the user to enter a string and then prints it in a fancy
"""

def symbol(text):
    if len(text) % 2 == 0:
        return "*"
    else:
        return "-"

def fancify(text):
    sym = symbol(text)
    text = sym + text + sym
    return text

def main():
    text = input("Enter a string: ")
    fancy_text = fancify(text)
    print("The fancy text is", fancy_text)
```

fancify.py 19:43 LF N UTF-8 Python 0 files

Function stack diagram



fancify.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom

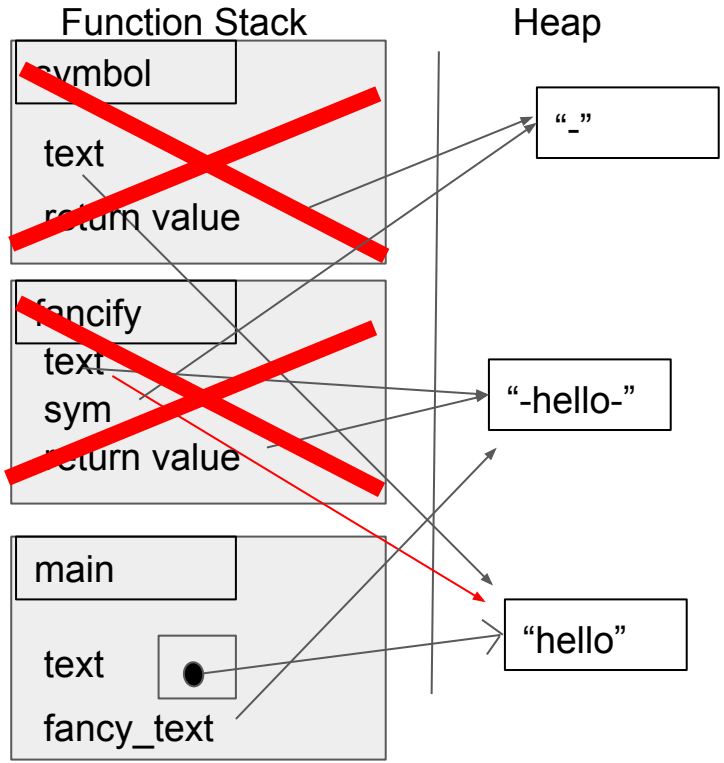
File Edit View Selection Find Packages Help

add.py | add2.py | add3.py | tip.py | fancify.py

```
"""
Asks the user to enter a string and then prints it in a fancy
"""

def symbol(text):
    if len(text) % 2 == 0:
        return "*"
    else:
        return "-"

def fancify(text):
    sym = symbol(text)
    return text

def main():
    text = input("Enter a string: ")
    fancy_text = fancify(text)
    print("The fancy text is", fancy_text)
```

fancify.py 19:43 LF N UTF-8 Python 0 files

Function stack diagram



fancify.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom

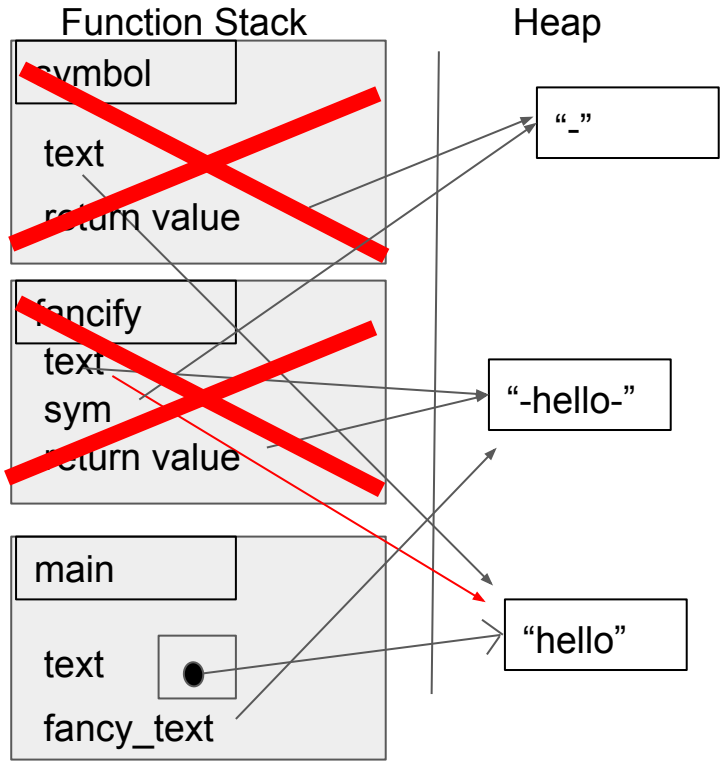
File Edit View Selection Find Packages Help

add.py | add2.py | add3.py | tip.py | fancify.py

```
"""
Asks the user to enter a string and then prints it in a fancy
"""

def symbol(text):
    if len(text) % 2 == 0:
        return "*"
    else:
        return "-"

def fancify(text):
    sym = symbol(text)
    text = sym + text + sym
    return text

def main():
    text = input("Enter a string: ")
    fancy_text = fancify(text)
    print("The fancy text is", fancy_text)
```

fancify.py 19:43 LF N UTF-8 Python 0 files

Function stack diagram



fancify.py — ~/CS21/cs21-devel/examples/inclass/w05 — Atom

File Edit View Selection Find Packages Help

add.py | add2.py | add3.py | tip.py | fancify.py

```
"""
Asks the user to enter a string and then prints it in a fancy
"""

def symbol(text):
    if len(text) % 2 == 0:
        return "*"
    else:
        return "-"

def fancify(text):
    sym = symbol(text)
    text = sym + text + sym
    return text

def main():
    text = input("Enter a string: ")
    fancy_text = fancify(text)
    print("The fancy text is", fancy_text)
```

fancify.py 19:43 LF N UTF-8 Python 0 files