

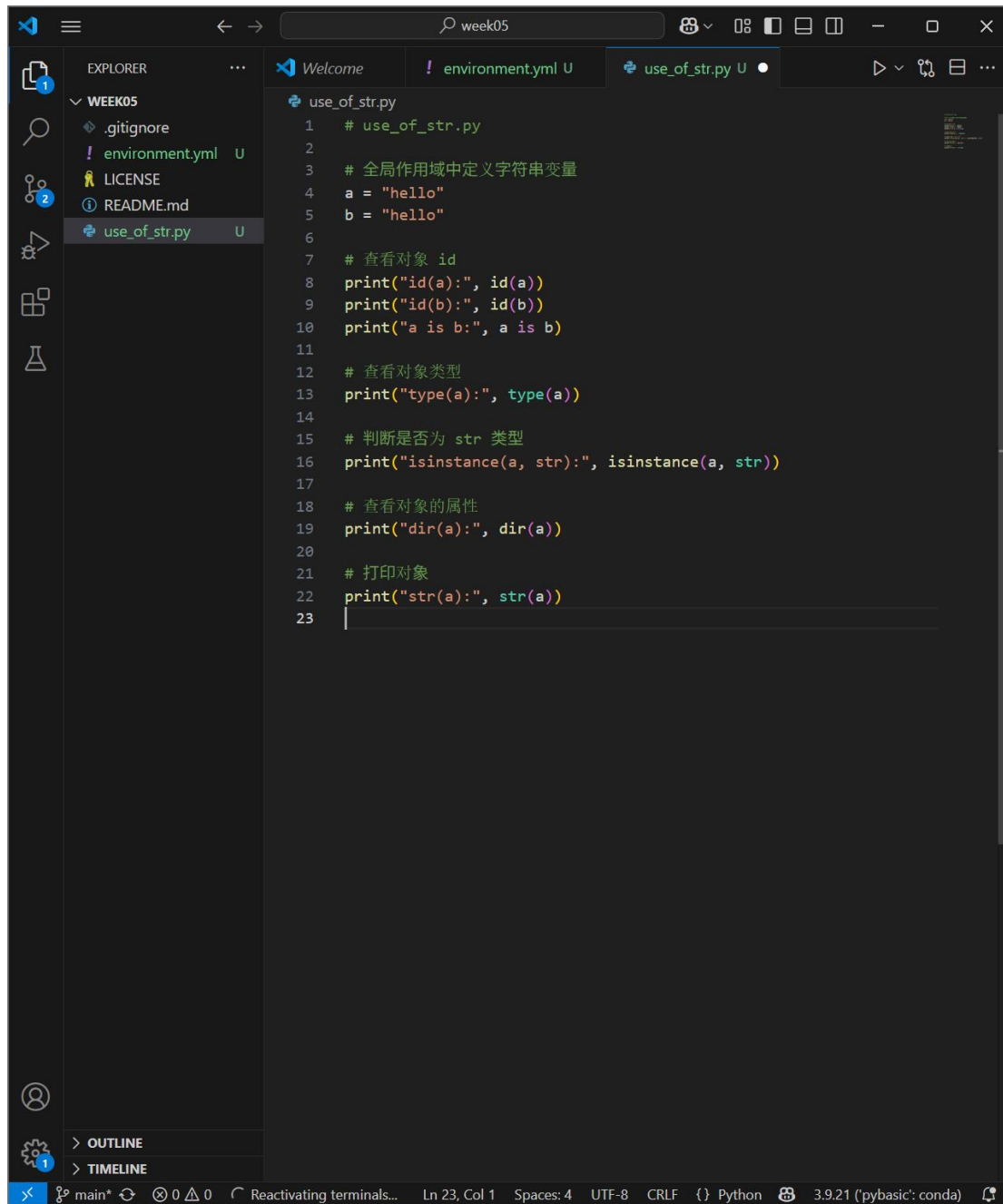
```
MINGW64:/c/Users/liu

liu@liujin MINGW64 ~ (master)
$ cd /c/Users/liu
git clone git@gitcode.com:liujin196/week05.git
Cloning into 'week05'...
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 5 (delta 0), reused 5 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (5/5), 8.44 KiB | 786.00 KiB/s, done.

liu@liujin MINGW64 ~ (master)
$ conda env create -f environment.yml
Retrieving notices: done

CondaValueError: prefix already exists: C:\Users\liu\anaconda3\envs\pybasic

liu@liujin MINGW64 ~ (master)
$
```



Visual Studio Code interface showing a Python script named `use_of_str.py` and its execution output in a terminal window.

EXPLORER:

- WEEK05
 - .gitignore
 - environment.yml U
 - LICENSE
 - README.md
 - use_of_str.py U

use_of_str.py > ...

```
1 # use_of_str.py
2
3 # 全局作用域中定义字符串变量
4 a = "hello"
5 b = "hello"
6
7 # 查看对象 id
8 print("id(a):", id(a))
9 print("id(b):", id(b))
10 print("a is b:", a is b)
11
12 # 查看对象类型
13 print("type(a):", type(a))
14
15 # 判断是否为 str 类型
16 print("isinstance(a, str):", isinstance(a, str))
17
18 # 查看对象的属性
19 print("dir(a):", dir(a))
20
21 # 打印对象
22 print("str(a):", str(a))
23
```

Terminal Output (MINGW64: c/Users/liu/week05):

```
liu@liujin MINGW64 ~/week05 (main)
$ python use_of_str.py
id(a): 1984703128944
id(b): 1984703128944
a is b: True
type(a): <class 'str'>
isinstance(a, str): True
dir(a): ['__add__', '__class__', '__contains__', '__delattr__', '__dir__', '__doc__', '__eq__', '__format__', '__ge__', '__getattr__', '__getitem__', '__getnewargs__', '__gt__', '__hash__', '__init__', '__init_subclass__', '__iter__', '__le__', '__len__', '__lt__', '__mod__', '__mul__', '__ne__', '__new__', '__reduce__', '__reduce_ex__', '__repr__', '__rmod__', '__rmul__', '__setattr__', '__sizeof__', '__str__', '__subclasshook__', 'capitalize', 'casefold', 'center', 'count', 'encode', 'endswith', 'expandtabs', 'find', 'format', 'format_map', 'index', 'isalnum', 'isalpha', 'isascii', 'isdecimal', 'isdigit', 'isidentifier', 'islower', 'isnumeric', 'isprintable', 'isspace', 'istitle', 'isupper', 'join', 'ljust', 'lower', 'lstrip', 'maketrans', 'partition', 'removeprefix', 'removesuffix', 'replace', 'rfind', 'rindex', 'rjust', 'rpartition', 'rsplit', 'rstrip', 'split', 'splitlines', 'startswith', 'strip', 'swapcase', 'title', 'translate', 'upper', 'zfill']
str(a): hello
(pybasic)
liu@liujin MINGW64 ~/week05 (main)
$
```

Bottom Bar: main* 0 0 0 Ln 23, Col 1 Spaces: 4 UTF-8 CRLF {} Python 3.9.21 (pybasic: conda)

Visual Studio Code interface showing a Python script and its execution output.

EXPLORER

- WEEK05
 - .gitignore
 - environment.yml U
 - LICENSE
 - README.md
 - use_of_int.py U
 - use_of_str.py U

use_of_int.py

```
1 # 全局作用域中定义整数变量
2 a = 100
3 b = 100
4
5 # 查看对象 id
6 print("id(a):", id(a))
7 print("id(b):", id(b))
8 print("a is b:", a is b)
9
10 # 查看对象类型
11 print("type(a):", type(a))
12
13 # 判断是否为 int 类型
14 print("isinstance(a, int):", isinstance(a, int))
15
16 # 查看对象的属性
17 print("dir(a):", dir(a))
18
19 # 打印对象
20 print("str(a):", str(a))
21
```

Terminal Output (MINGW64):

```
(pybasic)
liu@liujin MINGW64 ~/week05 (main)
$ python use_of_int.py
id(a): 1794849986000
id(b): 1794849986000
a is b: True
type(a): <class 'int'>
isinstance(a, int): True
dir(a): ['__abs__', '__add__', '__and__', '__bool__', '__ceil__', '__class__',
'__delattr__', '__dir__', '__divmod__', '__doc__', '__eq__', '__float__', '__floordiv__', '__format__', '__ge__', '__getattribute__', '__getnewargs__', '__gt__', '__hash__', '__index__', '__init__', '__init_subclass__', '__int__', '__invert__', '__le__', '__lshift__', '__lt__', '__mod__', '__mul__', '__neg__', '__new__', '__or__', '__pos__', '__pow__', '__radd__', '__rand__', '__rdivmod__', '__reduce__', '__reduce_ex__', '__repr__', '__rfloordiv__', '__rshift__', '__rsub__', '__rmul__', '__ror__', '__round__', '__rpow__', '__rrshift__', '__rshift__', '__sub__', '__subclasshook__', '__rxor__', '__setattr__', '__sizeof__', '__str__', '__sub__', '__subclasshook__', '__truediv__', '__trunc__', '__xor__', 'as_integer_ratio', 'bit_length', 'conjugate', 'denominator', 'from_bytes', 'imag', 'numerator', 'real', 'to_bytes']
str(a): 100
(pybasic)
liu@liujin MINGW64 ~/week05 (main)
$
```

Status Bar: main* 0 0 0 Ln 21, Col 1 Spaces: 4 UTF-8 CRLF {} Python 3.9.21 (pybasic: conda)

Visual Studio Code interface showing a Python file named `use_of_float.py` and its execution output in a terminal window.

EXPLORER

- WEEK05
 - .gitignore
 - environment.yml
 - LICENSE
 - README.md
 - use_of_float.py
 - use_of_int.py
 - use_of_str.py

use_of_float.py

```
1 # 全局作用域中定义浮点数变量
2 a = 3.14
3 b = 3.14
4
5 # 查看对象 id
6 print("id(a):", id(a))
7 print("id(b):", id(b))
8 print("a is b:", a is b)
9
10 # 查看对象类型
11 print("type(a):", type(a))
12
13 # 判断是否为 float 类型
14 print("isinstance(a, float):", isinstance(a, float))
15
16 # 查看对象的属性
17 print("dir(a):", dir(a))
18
19 # 打印对象
20 print("str(a):", str(a))
21
```

Terminal Output (MINGW64/c/Users/liu/week05)

```
tes', 'imag', 'numerator', 'real', 'to_bytes']
str(a): 100
(pybasic)
liu@liujin MINGW64 ~/week05 (main)
$ python use_of_float.py
id(a): 2214658332848
id(b): 2214658332848
a is b: True
type(a): <class 'float'>
isinstance(a, float): True
dir(a): ['__abs__', '__add__', '__bool__', '__ceil__', '__class__', '__delattr__', '__dir__', '__divmod__', '__doc__', '__eq__', '__float__', '__floor__', '__floordiv__', '__format__', '__ge__', '__getattribute__', '__getformat__', '__getnewargs__', '__gt__', '__hash__', '__init__', '__init_subclass__', '__int__', '__le__', '__lt__', '__mod__', '__mul__', '__ne__', '__neg__', '__new__', '__pos__', '__pow__', '__radd__', '__rdivmod__', '__reduce__', '__reduce_ex__', '__repr__', '__rfloordiv__', '__rmod__', '__rmul__', '__round__', '__rpow__', '__rsub__', '__rtruediv__', '__setattr__', '__setformat__', '__sizeof__', '__str__', '__subclasshook__', '__truediv__', '__trunc__', '__as_integer_ratio__', 'conjugate', 'fromhex', 'hex', 'imag', 'is_integer', 'real']
str(a): 3.14
(pybasic)
liu@liujin MINGW64 ~/week05 (main)
$
```

STATUS BAR

main* 0 0 0 Ln 21, Col 1 Spaces: 4 UTF-8 CRLF {} Python 3.9.21 (pybasic: conda)

Visual Studio Code interface showing a Python file named `use_of_bool.py` and its execution output in a terminal window.

EXPLORER

- WEEK05
 - .gitignore
 - environment.yml
 - LICENSE
 - README.md
 - use_of_bool.py
 - use_of_float.py
 - use_of_int.py
 - use_of_str.py

use_of_bool.py

```
1 # 全局作用域中定义布尔变量
2 a = True
3 b = True
4
5 # 查看对象 id
6 print("id(a):", id(a))
7 print("id(b):", id(b))
8 print("a is b:", a is b)
9
10 # 查看对象类型
11 print("type(a):", type(a))
12
13 # 判断是否为 bool 类型
14 print("isinstance(a, bool):", isinstance(a, bool))
15
16 # 查看对象的属性
17 print("dir(a):", dir(a))
18
19 # 打印对象
20 print("str(a):", str(a))
21
```

Terminal Output (MINGW64: c/Users/liu/week05)

```
(pybasic)
liu@liujin MINGW64 ~/week05 (main)
$ python use_of_bool.py
id(a): 140709470500968
id(b): 140709470500968
a is b: True
type(a): <class 'bool'>
isinstance(a, bool): True
dir(a): ['__abs__', '__add__', '__and__', '__bool__', '__ceil__', '__class__', '__delattr__', '__dir__', '__divmod__', '__doc__', '__eq__', '__float__', '__floordiv__', '__format__', '__ge__', '__getattr__', '__getnewargs__', '__gt__', '__hash__', '__index__', '__init__', '__init_subclass__', '__int__', '__invert__', '__le__', '__lshift__', '__lt__', '__mod__', '__mul__', '__neg__', '__new__', '__or__', '__pos__', '__pow__', '__radd__', '__rand__', '__rdivmod__', '__reduce__', '__reduce_ex__', '__repr__', '__rfloordiv__', '__lshift__', '__rmod__', '__rmul__', '__ror__', '__round__', '__rpow__', '__rshift__', '__rshift__', '__rsub__', '__rtruediv__', '__rxor__', '__setattr__', '__sizeof__', '__str__', '__sub__', '__subclasshook__', '__truediv__', '__trunc__', '__xor__', '__as_integer_ratio__', 'bit_length', 'conjugate', 'denominator', 'from_bytes', 'imag', 'numerator', 'real', 'to_bytes']
str(a): True
(pybasic)
liu@liujin MINGW64 ~/week05 (main)
$
```

Bottom Bar

main* 0 0 0 Ln 21, Col 1 Spaces: 4 UTF-8 CRLF {} Python 3.9.21 (pybasic: conda)

Visual Studio Code interface showing a Python file named `use_of_list.py` and its execution output in a terminal window.

EXPLORER

- WEEK05
 - .gitignore
 - environment.yml
 - LICENSE
 - README.md
 - use_of_bool.py
 - use_of_float.py
 - use_of_int.py
 - use_of_list.py**
 - use_of_str.py

use_of_list.py

```
1 # 全局作用域中定义列表变量
2 a = [1, 2, 3]
3 b = [1, 2, 3]
4
5 # 查看对象 id
6 print("id(a):", id(a))
7 print("id(b):", id(b))
8 print("a is b:", a is b)
9
10 # 查看对象类型
11 print("type(a):", type(a))
12
13 # 判断是否为 list 类型
14 print("isinstance(a, list):", isinstance(a, list))
15
16 # 查看对象的属性
17 print("dir(a):", dir(a))
18
19 # 打印对象
20 print("str(a):", str(a))
21
```

Terminal Output (MINGW64: c/Users/liu/week05)

```
zeof', '__str__', '__sub__', '__subclasshook__', '__truediv__', '__trunc__',
__xor__', 'as_integer_ratio', 'bit_length', 'conjugate', 'denominator', 'from_by
tes', 'imag', 'numerator', 'real', 'to_bytes']
str(a): True
(pybasic)
liu@liujin MINGW64 ~/week05 (main)
$ python use_of_list.py
id(a): 1934165894656
id(b): 1934165897408
a is b: False
type(a): <class 'list'>
isinstance(a, list): True
dir(a): ['__add__', '__class__', '__class_getitem__', '__contains__', '__delattr
__', '__delitem__', '__dir__', '__doc__', '__eq__', '__format__', '__ge__', '__g
etattribute__', '__getitem__', '__gt__', '__hash__', '__iadd__', '__imul__', '__
init__', '__init_subclass__', '__iter__', '__le__', '__len__', '__lt__', '__mul
__', '__ne__', '__new__', '__reduce__', '__reduce_ex__', '__repr__', '__reversed
__', '__rmul__', '__setattr__', '__setitem__', '__sizeof__', '__str__', '__subcla
sshook__', 'append', 'clear', 'copy', 'count', 'extend', 'index', 'insert', 'pop
', 'remove', 'reverse', 'sort']
str(a): [1, 2, 3]
(pybasic)
liu@liujin MINGW64 ~/week05 (main)
$
```

STATUS BAR

main* 0 0 0 Ln 21, Col 1 Spaces: 4 UTF-8 CRLF {} Python 3.9.21 (pybasic: conda)

Visual Studio Code interface showing a Python file named `use_of_dict.py` and its execution output in a terminal window.

EXPLORER

- WEEK05
 - .gitignore
 - environment.yml U
 - LICENSE
 - README.md
 - use_of_bool.py U
 - use_of_dict.py U**
 - use_of_float.py U
 - use_of_int.py U
 - use_of_list.py U
 - use_of_str.py U

use_of_dict.py

```
1 # 全局作用域中定义字典变量
2 a = {"name": "liu", "age": 28}
3 b = {"name": "liu", "age": 28}
4
5 # 查看对象 id
6 print("id(a):", id(a))
7 print("id(b):", id(b))
8 print("a is b:", a is b)
9
10 # 查看对象类型
11 print("type(a):", type(a))
12
13 # 判断是否为 dict 类型
14 print("isinstance(a, dict):", isinstance(a, dict))
15
16 # 查看对象的属性
17 print("dir(a):", dir(a))
18
19 # 打印对象
20 print("str(a):", str(a))
21
```

Terminal Output (MINGW64: c/Users/liu/week05)

```
_, '_rmul_', '_setattr_', '_setitem_', '_sizeof_', '_str_', '_subcla
sshook_', 'append', 'clear', 'copy', 'count', 'extend', 'index', 'insert', 'pop
', 'remove', 'reverse', 'sort']
str(a): [1, 2, 3]
(pybasic)
liu@liujin MINGW64 ~/week05 (main)
$ python use_of_dict.py
id(a): 2110444240576
id(b): 2110444240768
a is b: False
type(a): <class 'dict'>
isinstance(a, dict): True
dir(a): ['__class__', '__class_getitem__', '__contains__', '__delattr__', '__del
item__', '__dir__', '__doc__', '__eq__', '__format__', '__ge__', '__getattribute
__', '__getitem__', '__gt__', '__hash__', '__init__', '__init_subclass__', '__io
r__', '__iter__', '__le__', '__len__', '__lt__', '__ne__', '__new__', '__or__',
__reduce__', '__reduce_ex__', '__repr__', '__reversed__', '__ror__', '__setattr
__', '__setitem__', '__sizeof__', '__str__', '__subclasshook__', 'clear', 'copy',
'fromkeys', 'get', 'items', 'keys', 'pop', 'popitem', 'setdefault', 'update',
'values']
str(a): {'name': 'liu', 'age': 28}
(pybasic)
liu@liujin MINGW64 ~/week05 (main)
$
```

Ln 21, Col 1 Spaces: 4 UTF-8 CRLF {} Python 3.9.21 (pybasic: conda)

Visual Studio Code interface showing a Python file named `use_of_tuple.py` and its execution output in a terminal window.

File Explorer (Left Panel):

- WEEK05
 - .gitignore
 - environment.yml
 - LICENSE
 - README.md
 - use_of_bool.py
 - use_of_dict.py
 - use_of_float.py
 - use_of_int.py
 - use_of_list.py
 - use_of_str.py
 - use_of_tuple.py

Code Editor (Main Panel):

```
1 # 全局作用域中定义元组变量
2 a = (1, 2, 3)
3 b = (1, 2, 3)
4
5 # 查看对象 id
6 print("id(a):", id(a))
7 print("id(b):", id(b))
8 print("a is b:", a is b)
9
10 # 查看对象类型
11 print("type(a):", type(a))
12
13 # 判断是否为 tuple 类型
14 print("isinstance(a, tuple):", isinstance(a, tuple))
15
16 # 查看对象的属性
17 print("dir(a):", dir(a))
18
19 # 打印对象
20 print("str(a):", str(a))
21
```

Terminal Window (Bottom Panel):

```
MINGW64: c:/Users/liu/week05
r_, '_iter_', '_le_', '_len_', '_lt_', '_ne_', '_new_', '_or_',
'_reduce_', '_reduce_ex_', '_repr_', '_reversed_', '_ror_', '_setattr_',
'_setitem_', '_sizeof_', '_str_', '_subclasshook_', '_clear_', '_copy_',
'_fromkeys_', '_get_', '_items_', '_keys_', '_pop_', '_popitem_', '_setdefault_', '_update_',
'_values_']
str(a): {'name': 'liu', 'age': 28}
(pybasic)
liu@liujin MINGW64 ~/week05 (main)
$ python use_of_tuple.py
id(a): 2581605567424
id(b): 2581605567424
a is b: True
type(a): <class 'tuple'>
isinstance(a, tuple): True
dir(a): ['__add__', '__class__', '__class_getitem__', '__contains__', '__delattr__',
'__dir__', '__doc__', '__eq__', '__format__', '__ge__', '__getattr__',
'__getitem__', '__getnewargs__', '__gt__', '__hash__', '__init__', '__init_subc
lass__', '__iter__', '__le_', '__len__', '__lt__', '__mul__', '__ne_', '__new_
eof_', '__str__', '__subclasshook__', 'count', 'index']
str(a): (1, 2, 3)
(pybasic)
liu@liujin MINGW64 ~/week05 (main)
$
```

Status Bar (Bottom):

main* 0 0 0 Ln 21, Col 1 Spaces: 4 UTF-8 CRLF {} Python 3.9.21 (pybasic: conda)

Visual Studio Code interface showing a Python file named `use_of_set.py` and its output in a terminal window.

EXPLORER

- WEEK05
 - .gitignore
 - environment.yml
 - LICENSE
 - README.md
 - use_of_bool.py
 - use_of_dict.py
 - use_of_float.py
 - use_of_int.py
 - use_of_list.py
 - use_of_set.py**
 - use_of_str.py
 - use_of_tuple.py

use_of_set.py

```
1 # 全局作用域中定义集合变量
2 a = {1, 2, 3}
3 b = {1, 2, 3}
4
5 # 查看对象 id
6 print("id(a):", id(a))
7 print("id(b):", id(b))
8 print("a is b:", a is b)
9
10 # 查看对象类型
11 print("type(a):", type(a))
12
13 # 判断是否为 set 类型
14 print("isinstance(a, set):", isinstance(a, set))
15
16 # 查看对象的属性
17 print("dir(a):", dir(a))
18
19 # 打印对象
20 print("str(a):", str(a))
21
```

Terminal Output (MINGW64):

```
eof __, __str__, __subclasshook__, 'count', 'index']
str(a): {1, 2, 3}
(pybasic)
liu@liujin MINGW64 ~/week05 (main)
$ python use_of_set.py
id(a): 2005989578560
id(b): 2005989577216
a is b: False
type(a): <class 'set'>
isinstance(a, set): True
dir(a): ['__and__', '__class__', '__class_getitem__', '__contains__', '__delattr__',
'__dir__', '__doc__', '__eq__', '__format__', '__ge__', '__getattr__',
'__gt__', '__hash__', '__iand__', '__init__', '__init_subclass__', '__ior__',
'__isub__', '__iter__', '__ixor__', '__le__', '__len__', '__lt__', '__ne__', '__n
ew__', '__or__', '__rand__', '__reduce__', '__reduce_ex__', '__repr__', '__ror
__', '__rsub__', '__rxor__', '__setattr__', '__sizeof__', '__str__', '__sub
__subclasshook__', '__xor__', 'add', 'clear', 'copy', 'difference', 'difference_u
pdate', 'discard', 'intersection', 'intersection_update', 'isdisjoint', 'issubse
t', 'issuperset', 'pop', 'remove', 'symmetric_difference', 'symmetric_difference
_update', 'union', 'update']
str(a): {1, 2, 3}
(pybasic)
liu@liujin MINGW64 ~/week05 (main)
$
```

STATUS BAR

main* 0 0 0 Ln 21, Col 1 Spaces: 4 UTF-8 CRLF {} Python 3.9.21 (pybasic: conda)