

Лабораторная работа №1

1. Файл myremotemodule.py

```
1 def myfoo():
2     author = "Jamilya"
3     print(f"{author}'s module is imported")
```

2. Файл activation_script.py

```
1 import re
2 import sys
3 from importlib.abc import PathEntryFinder
4 from importlib.util import spec_from_loader
5 from urllib.request import urlopen
6
7
8 class URLFinder(PathEntryFinder):
9     def __init__(self, url, available):
10         self.url = url
11         self.available = available
12
13     def find_spec(self, name, target=None):
14         if name in self.available:
15             origin = "{}/{}.py".format(*args: self.url, name)
16             loader = URLLoader()
17             return spec_from_loader(name, loader, origin=origin)
18
19         else:
20             return None
21
22
23 def url_hook(some_str):
24     if not some_str.startswith(("http", "https")):
25         raise ImportError
26     with urlopen(some_str) as page: # requests.get()
27         data = page.read().decode("utf-8")
28         filenames = re.findall(pattern: "[a-zA-Z_][a-zA-Z0-9_]*.py", data)
29         modnames = {name[:-3] for name in filenames}
30         return URLFinder(some_str, modnames)
31
32
33 sys.path_hooks.append(url_hook)
34 print(sys.path_hooks)
35
36
37 class URLLoader:
38     def create_module(self, target):
39         return None
40
41     def exec_module(self, module):
42         with urlopen(module.__spec__.origin) as page:
43             source = page.read()
44             code = compile(source, module.__spec__.origin, mode="exec")
45             exec(code, module.__dict__)
```

3. Запуск сервера

```
(.venv) PS D:\PycharmProjects\Prog5\LR1> cd rootserver
(.venv) PS D:\PycharmProjects\Prog5\LR1\rootserver> python3 -m http.server
Serving HTTP on :: port 8000 (http://[::]:8000/) ...
```

4. Запуск файла activation_script.py

```
(.venv) PS D:\PycharmProjects\Prog5\LR1> python3 -i activation_script.py  
[<class 'zipimport.zipimporter'>, <function FileFinder.path_hook.<locals>.path_hook_for_FileFinder at 0x0000022687940EA0>, <function url_hook at 0x0000022687DB4540>]
```

5. Попытка импорта модуля

```
>>> import myremotemodule  
Traceback (most recent call last):  
  File "<stdin>", line 1, in <module>  
ModuleNotFoundError: No module named 'myremotemodule'  
>>>
```

6. Добавление пути ,где располагается модуль

```
>>> sys.path.append("http://localhost:8000")  
>>>
```



7. Импорт модуля

```
>>> import myremotemodule  
>>> myremotemodule  
<module 'myremotemodule' (http://localhost:8000/myremotemodule.py)>  
>>> myremotemodule.myfoo()  
Jamilya's module is imported  
>>>
```

8. Repl.it

Myremotemodule.py

```
myremotemodule.py > f myfoo > ...  
1 def myfoo():  
2     author = "Jamilya"  
3     print(f"{author}'s module is imported")  
4     return f"Модуль от {author} успешно загружен!"  
5  
6 MODULE_VERSION = "1.0"  
7  
8 if __name__ == "__main__":  
9     print("Этот модуль предназначен для удаленного импорта")  
10
```

Urlhook.py

```
urlhook.py > URLLoader > f exec_module > ...  
1 import re  
2 import sys  
3 import requests  
4 from importlib.abc import PathEntryFinder  
5 from importlib.util import spec_from_loader  
6  
7 class URLLoader:  
8     def create_module(self, target):  
9         return None  
10  
11     def exec_module(self, module):  
12         try:  
13             print(f"Загружаем модуль из: {module.__spec__.origin}")  
14             response = requests.get(module.__spec__.origin)  
15             response.raise_for_status()  
16             source = response.text  
17             code = compile(source, module.__spec__.origin, mode="exec")  
18             exec(code, module.__dict__)  
19             print("Модуль успешно загружен!")  
20         except requests.exceptions.RequestException as e:  
21             raise ImportError(f"Не удалось загрузить модуль: {e}") from e  
22  
23 class URLFinder(PathEntryFinder):  
24     def __init__(self, url, available):  
25         self.url = url  
26         self.available = available  
27  
28     def find_spec(self, name, target=None):  
29         if name in self.available:  
30             origin = "{}/{}.py".format(self.url, name)  
31             loader = URLLoader()  
32             return spec_from_loader(name, loader, origin=origin)  
33         else:  
34             return None  
35  
36 def url_hook(some_str):  
37     if not some_str.startswith(("http", "https")):  
38         raise ImportError  
39  
40     try:  
41         print(f"🔍 Проверяем хост: {some_str}")  
42         response = requests.get(some_str)  
43         response.raise_for_status()  
44         data = response.text  
45         filenames = re.findall("[a-zA-Z][a-zA-Z0-9_]*.py", data)  
46         modnames = {name[:-3] for name in filenames}  
47         print(f"Найдены модули: {modnames}")  
48         return URLFinder(some_str, modnames)  
49     except requests.exceptions.RequestException as e:  
50         print(f"Хост {some_str} недоступен: {e}")  
51         raise ImportError(f"Хост недоступен: {some_str}") from e  
52  
53 # Добавляем хук в sys.path_hooks  
54 sys.path_hooks.append(url_hook)  
55 print("URL hook активирован!")  
56
```

Test.py

```
test.py > ...
1 import sys
2
3 # Сначала активируем хук
4 import urlhook
5
6 print("=== ТЕСТ УДАЛЕННОГО ИМПОРТА ===")
7
8 # 1. Пытаемся импортировать без добавления пути
9 print("\n1. Попытка импорта без URL пути:")
10 try:
11     import myremotemodule
12     print("✅ Успешно!")
13 except ModuleNotFoundError as e:
14     print(f"❌ Ожидаемая ошибка: {e}")
15
16 # 2. Добавляем URL путь
17 print("\n2. Добавляем URL путь:")
18 sys.path.append("http://localhost:8000")
19 print("✅ Путь http://localhost:8000 добавлен в sys.path")
20
21 # 3. Пытаемся импортировать с URL путем
22 print("\n3. Попытка импорта с URL путем:")
23 try:
24     import myremotemodule
25     print("✅ Модуль успешно импортирован удаленно!")
26
27     # 4. Тестируем функции модуля
28     print("\n4. Тестируем функции модуля:")
29     result = myremotemodule.myfoo()
30     print(f"Результат: {result}")
31     print(f"Версия модуля: {myremotemodule.MODULE_VERSION}")
32
33 except Exception as e:
34     print(f"❌ Ошибка при импорте: {e}")
35
36 # 5. Тестируем обработку ошибок
37 print("\n5. Тестируем обработку недоступного хоста:")
38 try:
39     sys.path.append("http://nonexistent-domain-12345.com")
40     import some_module
41 except ImportError as e:
42     print(f"✅ Обработка ошибок работает: {e}")
43
```

```
~/workspace$ python3 -m http.server 8000
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
127.0.0.1 - - [10/Sep/2025 09:50:05] "GET / HTTP/1.1" 200 -
```

```
~/workspace: python3 test.py
~/workspace$ python3 test.py
URL hook активирован!
=== ТЕСТ УДАЛЕННОГО ИМПОРТА ===
1. Попытка импорта без URL пути:
✅ Успешно!

2. Добавляем URL путь:
✅ Путь http://localhost:8000 добавлен в sys.path

3. Попытка импорта с URL путем:
✅ Модуль успешно импортирован удаленно!

4. Тестируем функции модуля:
Jamliya's module is imported
Результат: Модуль от Jamliya успешно загружен!
Версия модуля: 1.0

5. Тестируем обработку недоступного хоста:
🔍 Проверим хост: http://localhost:8000
Найдены модули: {'activation_script', '.', 'main', 'urlhook', 'myremotemodule', 'test'}
🔍 Проверим хост: http://nonexistent-domain-12345.com
Хост http://nonexistent-domain-12345.com недоступен: HTTPConnectionPool(host='nonexistent-domain-12345.com', port=80): Max retries exceeded with url: / (Caused by NameResolutionError("<url1
lib3.connection.HTTPConnection object at 0x7f266b9ed9d0>: Failed to resolve 'nonexistent-domain-12345.com' ([Errno -2] Name or service not known)")
✅ Обработка ошибок работает: No module named 'some_module'
```

Generate (2/1)