



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение  
высшего образования

«МИРЭА – Российский технологический университет»

**РТУ МИРЭА**

---

---

**Институт информационных технологий (ИИТ)**  
**Кафедра математического обеспечения и стандартизации**  
**информационных технологий**

**ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ**  
по дисциплине «Технология разработки программных приложений»

**Практическое задание № 2**

Студент группы

ИНБО-08-22 Самойлов М.М.

\_\_\_\_\_  
(подпись)

Старший  
преподаватель

*Мельников Д. А.*

\_\_\_\_\_  
(подпись)

Отчет представлен

«13» марта 2024г.

Москва 2024 г

# ОГЛАВЛЕНИЕ

Задание.....	3
Выполнение заданий .....	4
Вывод .....	12

# Задание

## Часть 1

1. Напишите сценарий, который выводит дату, время, список зарегистрировавшихся пользователей, и uptime системы и сохраняет эту информацию в файл.
2. Напишите сценарий, который выводит содержимое любого каталога или сообщение о том, что его не существует.
3. Напишите сценарий, который с помощью цикла прочитает файл и выведет его содержимое.
4. Напишите сценарий, который с помощью цикла выведет список файлов и директорий из текущего каталога, укажет, что есть файл, а что директория.
5. Напишите сценарий, который подсчитает объем диска, занимаемого директорией. В качестве директории можно выбрать любую директорию в системе.
6. Напишите сценарий, который выведет список всех исполняемых файлов в директории, для которых у текущего пользователя есть права на исполнение.

## Часть 2

# Выполнение заданий

## Часть 1

```
$ task1.sh × $ task2.sh $ task3.sh $ task4.sh
Users > ryoichi > Documents > sem4 > trpp > pr2 > part1 > $ ta
1  #!/bin/bash
2  echo "Date and Time: $(date)" > system-info.txt
3  echo -e "Users:\n$(who)" >> system-info.txt
4  echo "Uptime: $(Uptime)" >> system-info.txt
```

Рисунок 1 – Скрипт для вывода даты, времени, имени пользователя и uptime

```
part1 — ryoichi@invoker — ..rpp/pr2/part1 — -zsh — 80x24
[→ part1 git:(main) × ./task1.sh
[→ part1 git:(main) × cat system-info.txt
Date and Time: Tue Mar 12 11:13:55 MSK 2024
Users:
ryoichi      console      Feb 18 11:16
ryoichi      ttys000      Mar 11 23:56
Uptime: 11:13 up 22 days, 23:57, 2 users, load averages: 2.52 2.78 2.99
USER      TTY      FROM      LOGIN@    IDLE WHAT
ryoichi   console  -          18Feb24  22days -
ryoichi   s000     -          Mon23     - Uptime
→ part1 git:(main) ×
```

Рисунок 2 – Результат выполнения первого скрипта

```
$ task2.sh × $ task3.sh $ task4.sh $ task5.sh
Users > ryoichi > Documents > sem4 > trpp > pr2 > part1 > $ task2.sh
1  #!/bin/bash
2  if [ -z "$1" ]; then
3  ls -l
4  elif [ -d "$1" ]; then
5  ls -l $1
6  else
7  echo "No such directory exists"
8  fi
9
```

Рисунок 3 – Скрипт для вывода содержимого каталога

```

[→ part1 git:(main) × ./task2.sh
total 64
-rw-r--r--  1 ryoichi  staff   23 Mar 12 00:23 file.txt
-rw-r--r--  1 ryoichi  staff  374 Mar 12 11:13 system-info.txt
-rwxr-xr-x@ 1 ryoichi  staff  147 Mar 12 10:21 task1.sh
-rwxr-xr-x@ 1 ryoichi  staff  111 Mar 12 10:21 task2.sh
-rwxr-xr-x@ 1 ryoichi  staff  138 Mar 12 10:21 task3.sh
drwxr-xr-x  2 ryoichi  staff   64 Mar 12 10:08 task4-dir
-rwxr-xr-x@ 1 ryoichi  staff  173 Mar 12 10:21 task4.sh
-rwxr-xr-x@ 1 ryoichi  staff   21 Mar 12 10:21 task5.sh
-rwxr-xr-x@ 1 ryoichi  staff   93 Mar 12 10:21 task6.sh
→ part1 git:(main) × █

```

Рисунок 4 – Результат выполнения второго скрипта

```

$ task3.sh × $ task4.sh $ task5.sh
Users > ryoichi > Documents > sem4 > trpp > pr2 >
1  #!/bin/bash
2  |
3  file=$1
4  if [ -f "$file" ]; then
5  |   while IFS= read -r var; do
6  | |   echo "$var"
7  | |   done < "$file"
8  else
9  |   echo "Bad argument!"
10 fi

```

Рисунок 5 – Скрипт для чтения и вывода содержимого файла

```

[→ part1 git:(main) × ./task3.sh
Bad argument!
[→ part1 git:(main) × ./task3.sh file.txt
some data
another data
→ part1 git:(main) × █

```

Рисунок 6 – Результат выполнения третьего скрипта

```
$ task4.sh × $ task5.sh $ task6.sh
Users > ryoichi > Documents > sem4 > trpp > pr2
1  #!/bin/bash
2
3  for var in $(ls)
4  do
5  if [ -f "$var" ]; then
6  echo "$var is a file"
7  elif [ -d "$var" ]; then
8  echo "$var is a directory"
9  else echo "$var is an unknown type"
10 fi
11 done
```

Рисунок 7 – Скрипт для вывода списка файлов и директорий с указанием типа

```
→ part1 git:(main) × ./task4.sh
file.txt is a file
system-info.txt is a file
task1.sh is a file
task2.sh is a file
task3.sh is a file
task4-dir is a directory
task4.sh is a file
task5.sh is a file
task6.sh is a file
→ part1 git:(main) ×
```

Рисунок 8 – Результат выполнения четвёртого скрипта

```
$ task5.sh × $ task6.sh
Users > ryoichi > Documents > sem4
1  #!/bin/bash
2  du -hs $1
```

Рисунок 9 – Скрипт для подсчёта объема диска

```
→ part1 git:(main) × ./task5.sh
32K .
→ part1 git:(main) × ./task5.sh task3.sh
4.0K task3.sh
→ part1 git:(main) ×
```

Рисунок 10 – Результат выполнения пятого скрипта

```
$ task6.sh ×
Users > ryoichi > Documents > sem4 > trpp > pr2 > part1
1  #!/bin/bash
2  for file in $(ls)
3  do
4  if [ ! -d $file ] && [ -x $file ]; then
5  echo "$file"
6  fi
7  done
```

Рисунок 11 – Скрипт для вывода списка исполняемых файлов

```
[→ part1 git:(main) × ./task6.sh
task1.sh
task2.sh
task3.sh
task4.sh
task5.sh
task6.sh
→ part1 git:(main) ×
```

Рисунок 12 – Результат выполнения шестого скрипта

## Часть 2

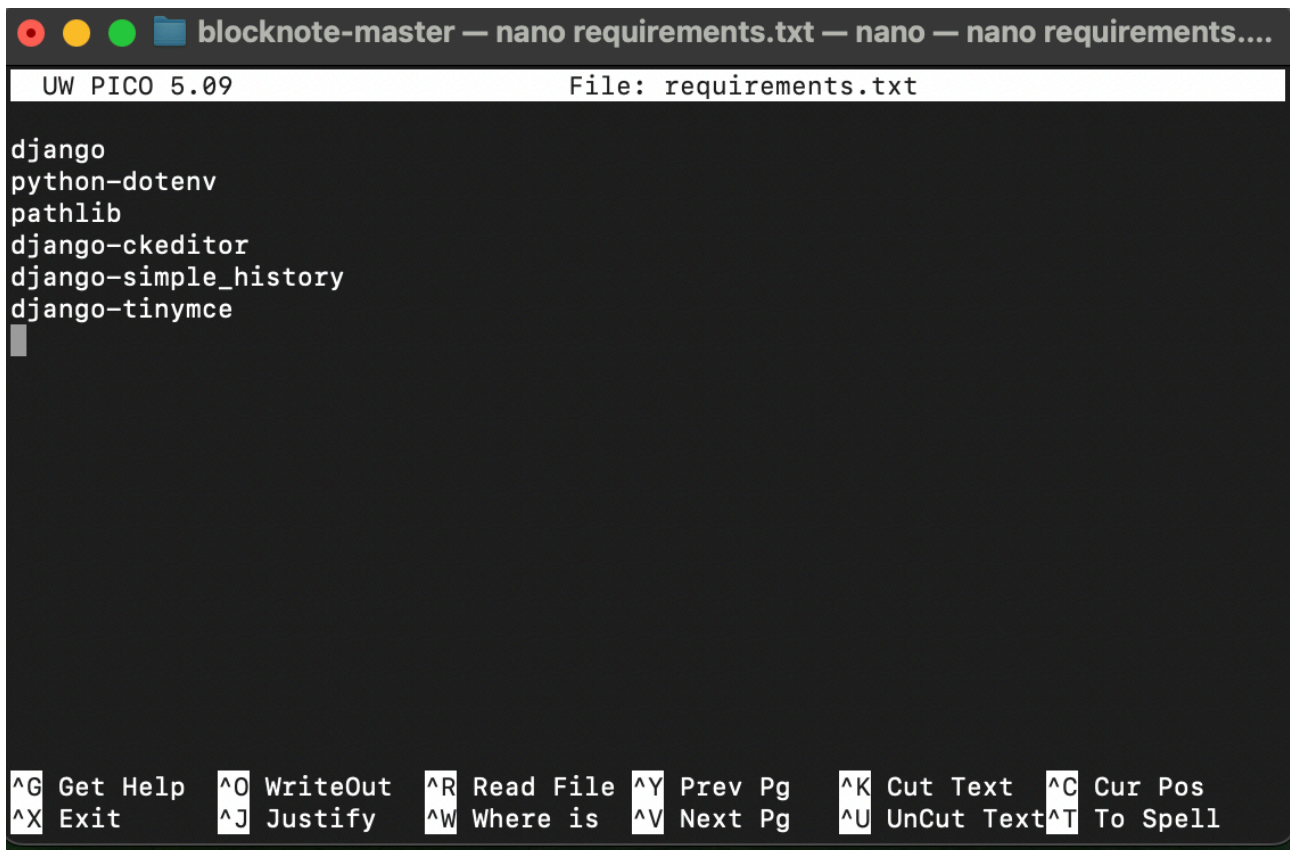
```
part2 — ryoichi@invoker — ..rpp/pr2/part2 — -zsh — 80x24
[→ part2 git:(main) ✕ mv /Users/ryoichi/Downloads/blocknote-master.tar.gz .
[→ part2 git:(main) ✕ tar -xvf blocknote-master.tar.gz
x blocknote-master/
x blocknote-master/.DS_Store
x blocknote-master/appengine/
x blocknote-master/.gitignore
x blocknote-master/static/
x blocknote-master/.gitattributes
x blocknote-master/templates/
x blocknote-master/manage.py
x blocknote-master/apps/
x blocknote-master/apps/.DS_Store
x blocknote-master/apps/todoapp/
x blocknote-master/apps/main/
x blocknote-master/apps/main/migrations/
x blocknote-master/apps/main/services.py
x blocknote-master/apps/main/models.py
x blocknote-master/apps/main/__init__.py
x blocknote-master/apps/main/apps.py
x blocknote-master/apps/main/forms.py
x blocknote-master/apps/main/admin.py
x blocknote-master/apps/main/tests.py
x blocknote-master/apps/main/urls.py
x blocknote-master/apps/main/views.py
```

Рисунок 13 – Скачивание и разархивация

```
[→ part2 git:(main) ✕ cd blocknote-master
[→ blocknote-master git:(main) ✕ ls
appengine apps      manage.py static      templates
[→ blocknote-master git:(main) ✕
```

Рисунок 14 – Содержимое папки



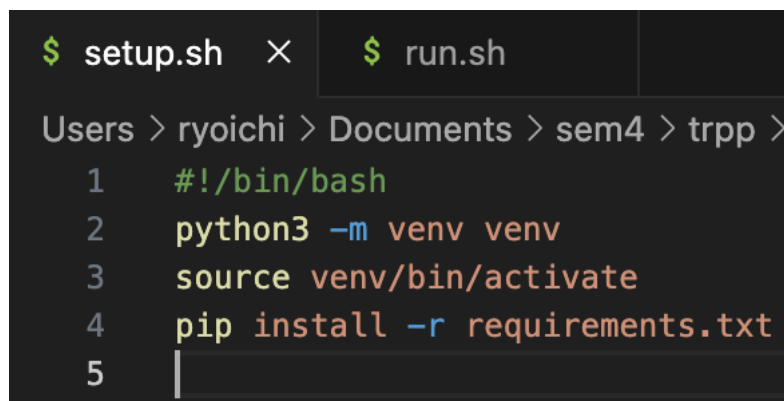


The screenshot shows a terminal window with the nano text editor open. The title bar reads "blocknote-master — nano requirements.txt — nano — nano requirements....". The editor's status bar at the top shows "UW PICO 5.09" and "File: requirements.txt". The main area of the editor contains the following text:

```
django
python-dotenv
pathlib
django-ckeditor
django-simple_history
django-tinymce
```

The bottom status bar of the nano editor displays various keyboard shortcuts: `^G` Get Help, `^O` WriteOut, `^R` Read File, `^Y` Prev Pg, `^K` Cut Text, `^C` Cur Pos, `^X` Exit, `^J` Justify, `^W` Where is, `^V` Next Pg, `^U` UnCut Text, and `^T` To Spell.

Рисунок 15 – Файл requirements.txt с зависимости



The screenshot shows a terminal window with two tabs: "setup.sh" and "run.sh". The "setup.sh" tab is active, showing the following script content:

```
Users > ryoichi > Documents > sem4 > trpp >
1  #!/bin/bash
2  python3 -m venv venv
3  source venv/bin/activate
4  pip install -r requirements.txt
5  |
```

Рисунок 16 – Скрипт setup

```
blocknote-master — ryoichi@invoker — ..cknote-master — -zsh — 80x24

Downloading django_js_asset-2.2.0-py3-none-any.whl.metadata (3.5 kB)
Downloading Django-5.0.3-py3-none-any.whl (8.2 MB)
 8.2/8.2 MB 13.7 MB/s eta 0:00:00
Downloading python_dotenv-1.0.1-py3-none-any.whl (19 kB)
Downloading pathlib-1.0.1-py3-none-any.whl (14 kB)
Downloading django_ckeditor-6.7.1-py3-none-any.whl (2.5 MB)
 2.5/2.5 MB 12.6 MB/s eta 0:00:00
Downloading django_simple_history-3.5.0-py3-none-any.whl (125 kB)
125.5/125.5 kB 10.7 MB/s eta 0:00:00
Downloading django_tinymce-3.7.1-py3-none-any.whl (1.1 MB)
 1.1/1.1 MB 16.0 MB/s eta 0:00:00
Downloading asgiref-3.7.2-py3-none-any.whl (24 kB)
Downloading django_js_asset-2.2.0-py3-none-any.whl (4.7 kB)
Downloading sqlparse-0.4.4-py3-none-any.whl (41 kB)
41.2/41.2 kB 3.0 MB/s eta 0:00:00
Installing collected packages: pathlib, sqlparse, python-dotenv, asgiref, django-
simple_history, django, django-tinymce, django-js-asset, django-ckeditor
Successfully installed asgiref-3.7.2 django-5.0.3 django-ckeditor-6.7.1 django-j
s-asset-2.2.0 django-simple_history-3.5.0 django-tinymce-3.7.1 pathlib-1.0.1 pyt
hon-dotenv-1.0.1 sqlparse-0.4.4

[notice] A new release of pip is available: 23.2.1 -> 24.0
[notice] To update, run: pip install --upgrade pip
→ blocknote-master git:(main) ×
```

Рисунок 17 – Запуск скрипта setup

```
$ run.sh ×
Users > ryoichi > Documents > sem4 > trpp > p
1  #!/bin/bash
2  source venv/bin/activate
3  python3 manage.py makemigrations
4  python3 manage.py migrate
5  python3 manage.py runserver
```

Рисунок 18 – Скрипт run

Проект содержит по пути /apps/todoapp/apps.py “Todo” код, который приводит к падению сборки. Для решения этой проблемы необходимо его очистить. Так же внутри проекта имеется импорт библиотеки `tinymce.widgets`, однако при использовании её в файле зависимостей происходит конфликт зависимостей. Это происходит из-за того, что данный проект создавался во времена использования уже устаревшей версии питона. Если убрать данную библиотеку в `requirements.txt`, то проблемы не будет.

```

r-4.24.0-LTS . You should consider strongly switching to a different editor (maybe CKEditor 5 respectively django-ckeditor-5 after checking whether the CKEditor 5 license terms work for you)
or switch to the non-free CKEditor 4 LTS package. See https://ckeditor.com/ckeditor-4-support/ for more on this. (Note! This notice has been added by the django-ckeditor developers and we are
not affiliated with CKSource and were not involved in the licensing change, so please refrain from complaining to us. Thanks.)
main.Article: (models.W042) Auto-created primary key used when not defining a primary key type, by default 'django.db.models.AutoField'.
    HINT: Configure the DEFAULT_AUTO_FIELD setting or the MainConfig.default_auto_field attribute to point to a subclass of AutoField, e.g. 'django.db.models.BigAutoField'.
main.GroupArticles: (models.W042) Auto-created primary key used when not defining a primary key type, by default 'django.db.models.AutoField'.
    HINT: Configure the DEFAULT_AUTO_FIELD setting or the MainConfig.default_auto_field attribute to point to a subclass of AutoField, e.g. 'django.db.models.BigAutoField'.
todoapp.Task: (models.W042) Auto-created primary key used when not defining a primary key type, by default 'django.db.models.AutoField'.
    HINT: Configure the DEFAULT_AUTO_FIELD setting or the AppConfig.default_auto_field attribute to point to a subclass of AutoField, e.g. 'django.db.models.BigAutoField'.
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, main, sessions, todoapp
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying admin.0003_logentry_add_action_flag_choices... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007_alter_validators_add_error_messages... OK
  Applying auth.0008_alter_user_username_max_length... OK
  Applying auth.0009_alter_user_last_name_max_length... OK
  Applying auth.0010_alter_group_name_max_length... OK
  Applying auth.0011_update_proxy_permissions... OK
  Applying auth.0012_alter_user_first_name_max_length... OK
  Applying auth.0013_historicaluser... OK
  Applying main.0001_initial... OK
  Applying main.0002_alter_historicalarticle_options_and_more... OK
  Applying sessions.0001_initial... OK
  Applying todoapp.0001_initial... OK
  Applying todoapp.0002_auto_20210316_1013... OK
Watching for file changes with StatReloader
Performing system checks...

System check identified some issues:

WARNINGS:
?: (ckeditor.W001) django-ckeditor bundles CKEditor 4.22.1 which isn't supported anymore and which does have unfixed security issues, see for example https://ckeditor.com/ckeditor-4-support/ or switch to the non-free CKEditor 4 LTS package. See https://ckeditor.com/ckeditor-4-support/ for more on this. (Note! This notice has been added by the django-ckeditor developers and we are
not affiliated with CKSource and were not involved in the licensing change, so please refrain from complaining to us. Thanks.)
main.Article: (models.W042) Auto-created primary key used when not defining a primary key type, by default 'django.db.models.AutoField'.
    HINT: Configure the DEFAULT_AUTO_FIELD setting or the MainConfig.default_auto_field attribute to point to a subclass of AutoField, e.g. 'django.db.models.BigAutoField'.
main.GroupArticles: (models.W042) Auto-created primary key used when not defining a primary key type, by default 'django.db.models.AutoField'.
    HINT: Configure the DEFAULT_AUTO_FIELD setting or the MainConfig.default_auto_field attribute to point to a subclass of AutoField, e.g. 'django.db.models.BigAutoField'.
todoapp.Task: (models.W042) Auto-created primary key used when not defining a primary key type, by default 'django.db.models.AutoField'.
    HINT: Configure the DEFAULT_AUTO_FIELD setting or the AppConfig.default_auto_field attribute to point to a subclass of AutoField, e.g. 'django.db.models.BigAutoField'.

System check identified 4 issues (0 silenced).
March 12, 2024 - 13:05:10
Django version 5.0.3, using settings 'appengine.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CONTROL-C.

```

Рисунок 19 – Запуск скрипта run

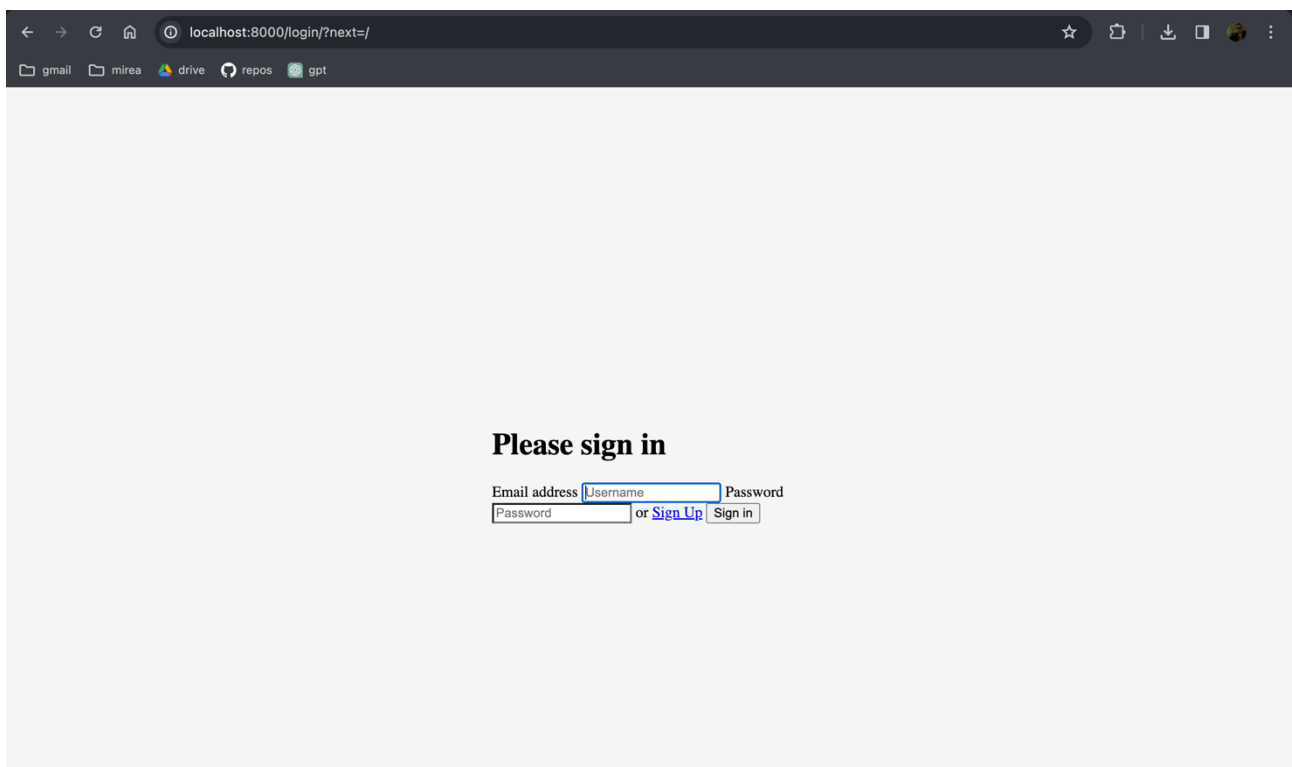


Рисунок 20 – Результат запуска

## **Вывод**

В ходе выполнения данной работы были изучены основы работы с Bash скриптами. Были написаны скрипты, выполняющие разные задачи, а также помогающие создать окружение, необходимое для запуска приложения.