

Лабораторная работа №1

Имитационное моделирование

Чистов Даниил Максимович

2026-02-21

Содержание I

1 Выполнение лабораторной работы

2 Выводы

3 Список литературы

Докладчик

- Чистов Даниил Максимович

Докладчик

- Чистов Даниил Максимович
- Студент

Докладчик

- Чистов Даниил Максимович
- Студент
- Третий курс

Докладчик

- Чистов Даниил Максимович
- Студент
- Третий курс
- Российский университет дружбы народов им. П. Лумумбы

Цель работы

- Целью данной работы является создание рабочего пространства для последующих работ, получение практических навыков с Julia и git.

Задание

- Создать рабочий каталог для всего курса.

Задание

- Создать рабочий каталог для всего курса.
- Создать рабочее пространство для программ в рамках лабораторной работы.

Задание

- Создать рабочий каталог для всего курса.
- Создать рабочее пространство для программ в рамках лабораторной работы.
- Установить необходимые пакеты.

Задание

- Создать рабочий каталог для всего курса.
- Создать рабочее пространство для программ в рамках лабораторной работы.
- Установить необходимые пакеты.
- Выполнить предложенный код.

Задание

- Создать рабочий каталог для всего курса.
- Создать рабочее пространство для программ в рамках лабораторной работы.
- Установить необходимые пакеты.
- Выполнить предложенный код.
- Преобразовать код в литературный стиль.

Задание

- Создать рабочий каталог для всего курса.
- Создать рабочее пространство для программ в рамках лабораторной работы.
- Установить необходимые пакеты.
- Выполнить предложенный код.
- Преобразовать код в литературный стиль.
- Сгенерировать из литературного кода: чистый код, jupyter notebook, документацию в формате Quarto.

Задание

- Создать рабочий каталог для всего курса.
- Создать рабочее пространство для программ в рамках лабораторной работы.
- Установить необходимые пакеты.
- Выполнить предложенный код.
- Преобразовать код в литературный стиль.
- Сгенерировать из литературного кода: чистый код, jupyter notebook, документацию в формате Quarto.
- Выполнить код из jupyter notebook.

Задание

- Создать рабочий каталог для всего курса.
- Создать рабочее пространство для программ в рамках лабораторной работы.
- Установить необходимые пакеты.
- Выполнить предложенный код.
- Преобразовать код в литературный стиль.
- Сгенерировать из литературного кода: чистый код, jupyter notebook, документацию в формате Quarto.
- Выполнить код из jupyter notebook.
- Интегрировать документацию в формате Quarto в отчёт.

Задание

- Создать рабочий каталог для всего курса.
- Создать рабочее пространство для программ в рамках лабораторной работы.
- Установить необходимые пакеты.
- Выполнить предложенный код.
- Преобразовать код в литературный стиль.
- Сгенерировать из литературного кода: чистый код, jupyter notebook, документацию в формате Quarto.
- Выполнить код из jupyter notebook.
- Интегрировать документацию в формате Quarto в отчёт.
- Добавить в код в литературном стиле вычисление для набора параметров.

Задание

- Создать рабочий каталог для всего курса.
- Создать рабочее пространство для программ в рамках лабораторной работы.
- Установить необходимые пакеты.
- Выполнить предложенный код.
- Преобразовать код в литературный стиль.
- Сгенерировать из литературного кода: чистый код, jupyter notebook, документацию в формате Quarto.
- Выполнить код из jupyter notebook.
- Интегрировать документацию в формате Quarto в отчёт.
- Добавить в код в литературном стиле вычисление для набора параметров.
- Сгенерировать из литературного кода: чистый код, jupyter notebook, документацию в формате Quarto.

Задание

- Создать рабочий каталог для всего курса.
- Создать рабочее пространство для программ в рамках лабораторной работы.
- Установить необходимые пакеты.
- Выполнить предложенный код.
- Преобразовать код в литературный стиль.
- Сгенерировать из литературного кода: чистый код, jupyter notebook, документацию в формате Quarto.
- Выполнить код из jupyter notebook.
- Интегрировать документацию в формате Quarto в отчёт.
- Добавить в код в литературном стиле вычисление для набора параметров.
- Сгенерировать из литературного кода: чистый код, jupyter notebook, документацию в формате Quarto.
- Выполнить код из jupyter notebook.

Задание

- Создать рабочий каталог для всего курса.
- Создать рабочее пространство для программ в рамках лабораторной работы.
- Установить необходимые пакеты.
- Выполнить предложенный код.
- Преобразовать код в литературный стиль.
- Сгенерировать из литературного кода: чистый код, jupyter notebook, документацию в формате Quarto.
- Выполнить код из jupyter notebook.
- Интегрировать документацию в формате Quarto в отчёт.
- Добавить в код в литературном стиле вычисление для набора параметров.
- Сгенерировать из литературного кода: чистый код, jupyter notebook, документацию в формате Quarto.
- Выполнить код из jupyter notebook.
- Интегрировать документацию в формате Quarto в отчёт.

Раздел 1

Выполнение лабораторной работы

Создание рабочего каталога для всего курса

Требуется создать рабочие каталоги на двух платформах - GitVerse и GitHub. Я воспользовался шаблоном курса, создал репозитории на GitVerse, аналогично на GitHub

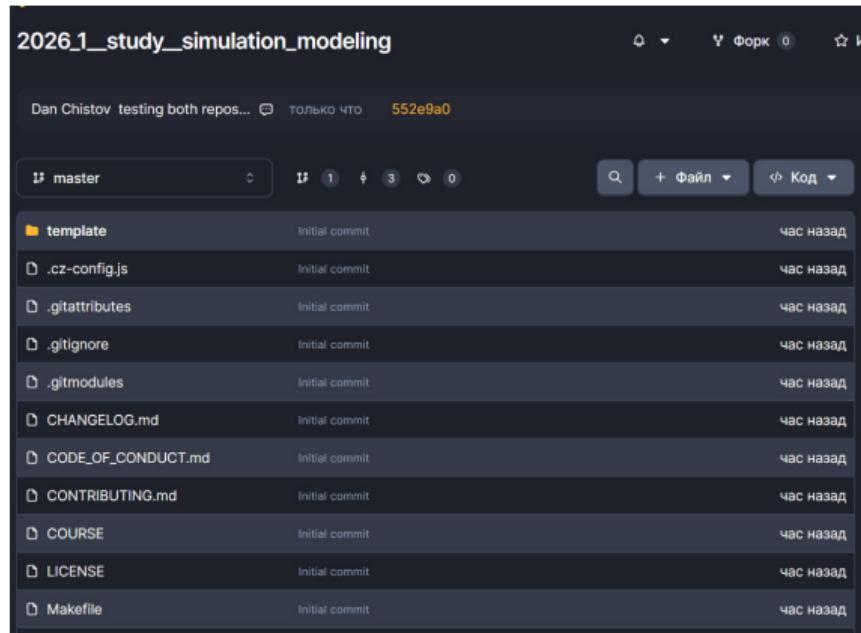


Рисунок 1: Созданный репозиторий на платформе GitVerse

Инициализация курса

Теперь перехожу в папку курса и инициализирую его командой make prepare.

```
l2232@Danya MSYS /c/Users/l2232/Documents/GitHub/2026-1--study--simulation-modeling
# echo simulation-modeling > COURSE

l2232@Danya MSYS /c/Users/l2232/Documents/GitHub/2026-1--study--simulation-modeling
# make prepare
Synchronizing submodule url for 'template/report'
Synchronizing submodule url for 'template/presentation'
Synchronizing submodule url for 'template/presentation'
Synchronizing submodule url for 'template/report'
```

Рисунок 2: Инициализация курса в консоли

Загрузка изменений

Загружаю изменения на сервер.

```
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling> git add .
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling> git commit -am 'feat(main): make course structure
'
[master f71efdd] feat(main): make course structure
 174 files changed, 6704 insertions(+), 259 deletions(-)
 delete mode 100644 CHANGELOG.md
 create mode 100644 labs/README.md
 create mode 100644 labs/README.ru.md
```

Рисунок 3: Загрузка изменений на сервер с помощью git

Отправление изменений

Отправляю изменения на обе платформы - GitHub (origin), GitVerse (simmod).

```
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling> git push origin master
Enumerating objects: 62, done.
Counting objects: 100% (62/62), done.
Delta compression using up to 8 threads
Compressing objects: 100% (46/46), done.
Writing objects: 100% (59/59), 701.22 KiB | 13.48 MiB/s, done.
Total 59 (delta 15), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (15/15), completed with 1 local object.
To https://github.com/danchist/2026-1--study--simulation-modeling.git
  552e9a0..f71efdd master -> master
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling> git push simmod master
Enumerating objects: 62, done.
Counting objects: 100% (62/62), done.
Delta compression using up to 8 threads
Compressing objects: 100% (46/46), done.
Writing objects: 100% (59/59), 701.22 KiB | 13.48 MiB/s, done.
Total 59 (delta 15), reused 0 (delta 0), pack-reused 0 (from 0)
remote: . Processing 1 references
remote: Processed 1 references in total
To https://gitverse.ru/danchist/2026_1__study__simulation_modeling.git
  552e9a0..f71efdd master -> master
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling>
```

Рисунок 4: Отправление изменений на GitHub и GitVerse



Создание рабочего каталога для всего курса

Изменения успешно внесены, рабочий каталог курса создан.

The screenshot shows a GitHub repository page for the repository "2026-1--study--simulation-modeling". The repository is public and was generated from [yamadharma/course-directory-student-template](#). The master branch has 1 branch and 0 tags. The commit history shows the following changes:

Commit	Message	Time
<code>feat(main): make course structure</code>	danchist	f71efdd · 1 minute ago
<code>labs</code>		feat(main): make course structure
<code>template</code>		Initial commit
<code>.cz-config.js</code>		Initial commit
<code>.gitattributes</code>		Initial commit
<code>.gitignore</code>		Initial commit
<code>.gitmodules</code>		Initial commit
<code>CODE_OF_CONDUCT.md</code>		Initial commit
<code>CONTRIBUTING.md</code>		Initial commit
<code>COURSE</code>		feat(main): make course structure
<code>LICENSE</code>		Initial commit
<code>Makefile</code>		Initial commit

Настройка git flow

По заданию курс требует git flow для работы. Инициализирую его, выбираю v, как префикс для новых версий.

```
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling> git flow init

Which branch should be used for bringing forth production releases?
  - master
Branch name for production releases: [master] master
Branch name for "next release" development: [develop] develop

How to name your supporting branch prefixes?
Feature branches? [feature/] feature
Bugfix branches? [bugfix/] bugfix/
Release branches? [release/] release/
Hotfix branches? [hotfix/] hotfix/
Support branches? [support/] support/
Version tag prefix? [] v
Hooks and filters directory? [C:/Users/12232/Documents/GitHub/2026-1--study--simulation-modeling/]
```

Рисунок 6: Инициализация git flow

Отправление изменений

После инициализации git flow появилось разделение на ветки develop и master, нужно внести изменения на обе платформы GitHub и GitVerse. Командой git branch проверяю, что я нахожусь на ветке develop. После чего отправляю изменения на платформы.

```
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling> git branch
* develop
  master
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling> git push -u --all
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'develop' on GitHub by visiting:
remote:     https://github.com/danchist/2026-1--study--simulation-modeling/pull/new/develop
remote:
To https://github.com/danchist/2026-1--study--simulation-modeling.git
 * [new branch]      develop -> develop
branch 'master' set up to track 'origin/master'.
branch 'develop' set up to track 'origin/develop'.
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling> git push -u simmod --all
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: You can create a PR using the link below:
remote:     https://gitverse.ru/danchist/2026_1__study__simulation_modeling/compare?baseBranch=master&headBranch=develop
remote:
remote: . Processing 1 references
remote: Processed 1 references in total
To https://gitverse.ru/danchist/2026_1__study__simulation_modeling.git
 * [new branch]      develop -> develop
branch 'master' set up to track 'simmod/master'.
branch 'develop' set up to track 'simmod/develop'.
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling>
```



Создание первого релиза

Инициализирую первый релиз - версия 1.0.0, затем командой standard-changelog --first-release создаю журнал изменений, добавляю изменения в git, создаю коммит.

```
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling> git flow release start 1.0.0
Switched to a new branch 'release/1.0.0'
```

Summary of actions:

- A new branch 'release/1.0.0' was created, based on 'develop'
- You are now on branch 'release/1.0.0'

Follow-up actions:

- Bump the version number now!
- Start committing last-minute fixes in preparing your release
- When done, run:

```
git flow release finish '1.0.0'
```

```
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling> standard-changelog --first-release
standard-changelog : Имя "standard-changelog" не распознано как имя командлета, функции, файла сценария или выполняемой программы.
Проверьте правильность написания имени, а также наличие и правильность пути, после чего повторите попытку.
```

строка:1 знак:1

```
+ standard-changelog --first-release
```

```
+ ~~~~~
```

```
  + CategoryInfo          : ObjectNotFound: (standard-changelog:String) [], CommandNotFoundException
  + FullyQualifiedErrorId : CommandNotFoundException
```

```
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling> npx standard-changelog --first-release
```

```
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling> git add CHANGELOG.md
```

```
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling> git commit -am 'chore(site): add changelog'
[release/1.0.0 f7fd693] chore(site): add changelog
```

Создание первого релиза

Отправляю изменения на обе платформы, включая созданный тэг.

```
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling> git push --all
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 8 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 699 bytes | 349.00 KiB/s, done.
Total 5 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: You can create a PR using the link below:
remote:   https://gitverse.ru/danchist/2026_1__study__simulation_modeling/compare?baseBranch=master&headBranch=develop
remote:
remote: .. Processing 2 references
remote: Processed 2 references in total
To https://gitverse.ru/danchist/2026_1__study__simulation_modeling.git
  f71efdd..37b06d6  develop -> develop
  f71efdd..9c39d65  master -> master
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling> git push --tags
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 176 bytes | 176.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote: . Processing 1 references
remote: Processed 1 references in total
To https://gitverse.ru/danchist/2026_1__study__simulation_modeling.git
 * [new tag]          v1.0.0 -> v1.0.0
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling> git push simmod --all
Everything up-to-date
```

Создание первого релиза

На GitVerse самостоятельно создаю первый релиз.

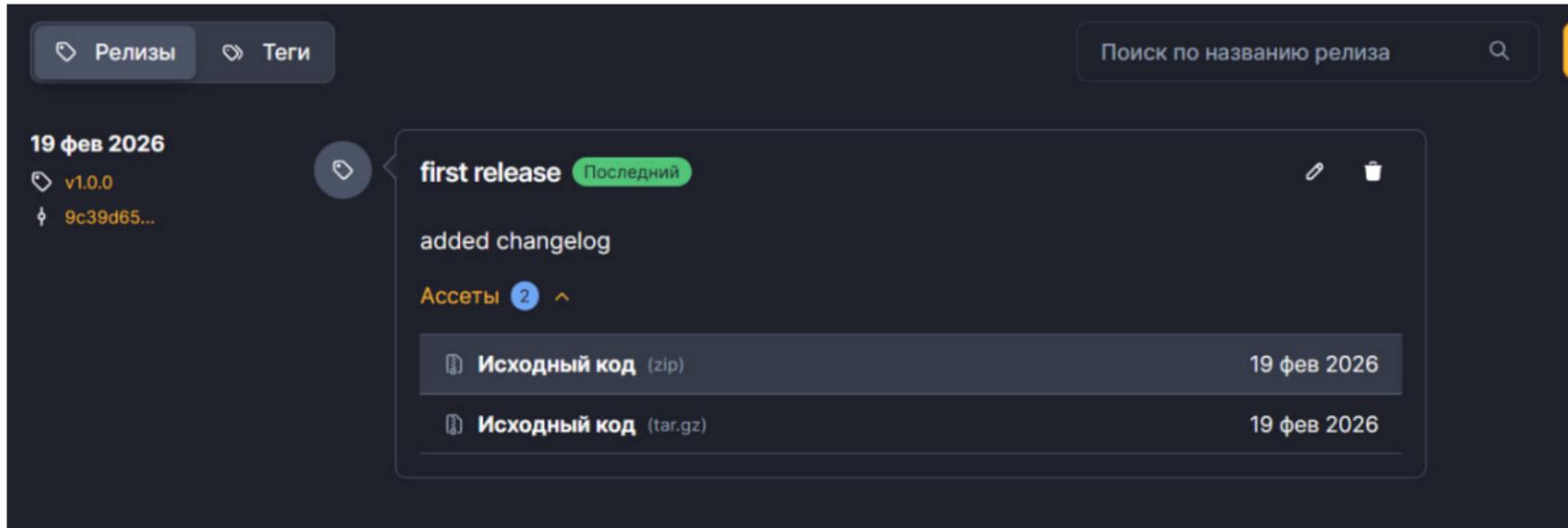


Рисунок 10: Первый релиз на GitVerse

Создание первого релиза

На GitHub создаю релиз командами `release create` и прикрепляю журнал изменений.

```
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling> cd release
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\release> gh release create v1.0.0 -F .\CHANGELOG.md
https://github.com/danchist/2026-1--study--simulation-modeling/releases/tag/v1.0.0
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\release>
```

The screenshot shows a GitHub release page for a repository named 'danchist'. The release is titled 'v1.0.0' and is marked as 'Latest'. Below the title, the release date '1.0.0 (2026-02-19)' is displayed. The 'Features' section lists one item: 'main: make course structure (f71efdd)'. The 'Assets' section contains two files: 'Source code (zip)' and 'Source code (tar.gz)', both uploaded 9 minutes ago. There is also a link to the repository's README at the bottom.

now

v1.0.0 Latest

danchist

v1.0.0

9c39d65

Compare ▾

1.0.0 (2026-02-19)

Features

- main: make course structure ([f71efdd](#))

Assets 2

Source code (zip)	9 minutes ago
Source code (tar.gz)	9 minutes ago

Создание проекта DrWatson для лабораторных работ

Перехожу в каталог lab01, первый раз запускаю Julia, после чего командами using Pkg, Pkg.add(«DrWatson») загружаю пакет DrWatson. Затем инициализирую проект.

```
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\release> cd ..
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling> cd .\labs\lab01\
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01> julia

julia> using Pkg

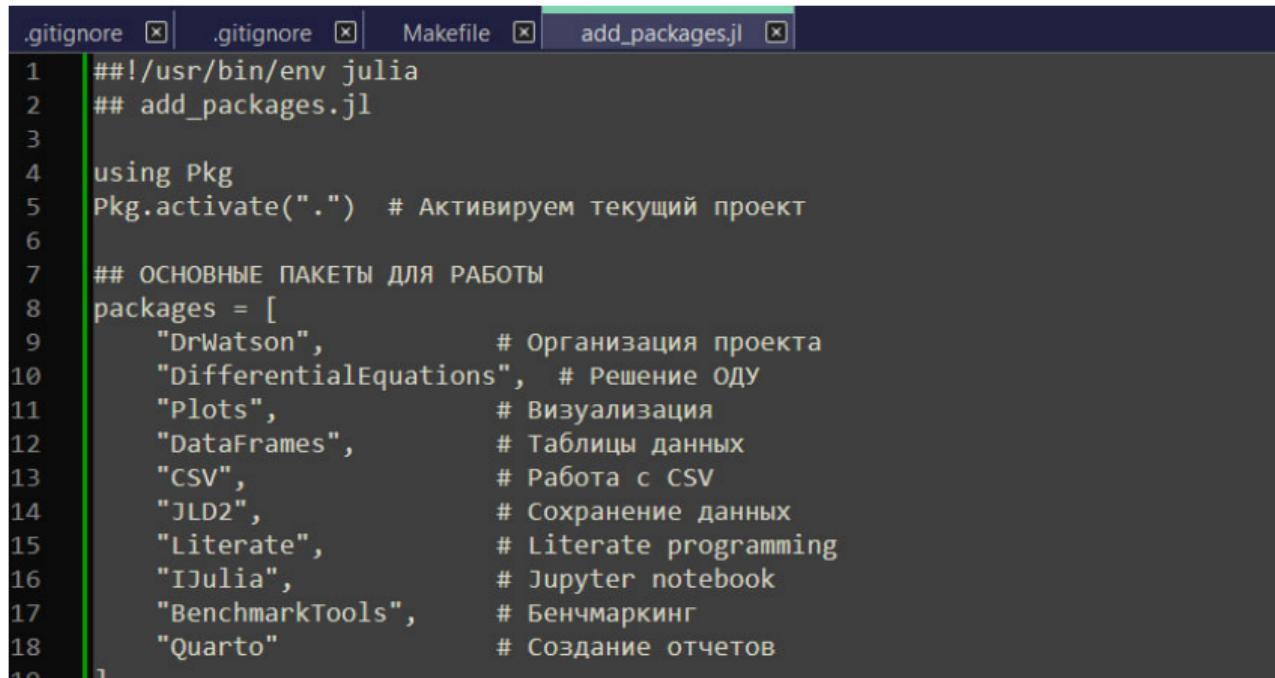
julia> Pkg.add("DrWatson")
Resolving package versions...
Project No packages added to or removed from `C:\Users\12232\.julia\environments\v1.12\Project.toml'
Manifest No packages added to or removed from `C:\Users\12232\.julia\environments\v1.12\Manifest.toml'

julia> using DrWatson

julia> initialize_project("project"; authors="Daniil Chistov", git=false)
Activating new project at `C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project'
Resolving package versions...
Updating `C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project\Project.toml`
```

Загрузка пакетов

Для дальнейшей работы требуется загрузить большее количество пакетов. Вместо того, чтобы делать это самостоятельно, воспользуюсь скриптом, который загрузит все требуемые пакеты. Ниже программный код данного скрипта.



```
1 ##!/usr/bin/env julia
2 ## add_packages.jl
3
4 using Pkg
5 Pkg.activate(".") # Активируем текущий проект
6
7 ## ОСНОВНЫЕ ПАКЕТЫ ДЛЯ РАБОТЫ
8 packages = [
9     "DrWatson",           # Организация проекта
10    "DifferentialEquations", # Решение ОДУ
11    "Plots",               # Визуализация
12    "DataFrames",          # Таблицы данных
13    "CSV",                  # Работа с CSV
14    "JLD2",                 # Сохранение данных
15    "Literate",             # Literate programming
16    "IJulia",                # Jupyter notebook
17    "BenchmarkTools",        # Бенчмаркинг
18    "Quarto"                 # Создание отчетов
19 ]
```

Загрузка пакетов

Запускаю скрипт, пакеты загружены успешно.

```
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project> julia .\add_packages.jl
Activating project at `C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project'
Установка базовых пакетов...
Resolving package versions...
Installed SciMLLogging — v1.9.0
Installed CSV — v0.10.16
Installed Plots — v1.41.6
Installed JumpProcesses — v9.22.0
Installing artifacts [6e4b80f9] + BenchmarkTools v1.6.3
[336ed68f] + CSV v0.10.16
[a93c6f00] + DataFrames v1.8.1
[0c46a032] + DifferentialEquations v7.17.0
[7073ff75] + IJulia v1.34.3
[033835bb] + JLD2 v0.6.3
[98b081ad] + Literate v2.21.0
[91a5bcdd] + Plots v1.41.6
[d7167be5] + Quarto v1.0.0
Updating `C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project\Manifest.toml`
[47edcb42] + ADTypes v1.21.0
[7d057e22] + Accessors v0.1.42
```

Рисунок 14: Загрузка требуемых пакетов

Проверка пакетов

Теперь требуется проверить загруженные пакеты следующим скриптом. Перед этим, в папке scripts создаю файл, вставляю требуемый программный код проверки пакетов.

The terminal window shows the user navigating through their GitHub repository to reach the 'scripts' directory. The Notepad++ window displays a Julia script named 'test_script.jl' with the following code:

```
#!/usr/bin/env julia
## test_setup.jl

using DrWatson
@quickactivate "project"

println("Проект активирован: ", projectdir())

## Проверка пакетов
packages = [
    "DrWatson",           # Организация проекта
    "DifferentialEquations", # Решение ОДУ
    "Plots",               # Визуализация
    "DataFrames",          # Таблицы данных
    "CSV",                  # Работа с CSV
    "JLD2",                 # Сохранение данных
    "Literate",              # Literate programming
    "IJulia",                # Jupyter notebook
    "BenchmarkTools",        # Бенчмаркинг
    "Quarto"                  # Создание отчетов
]

println("\nПроверка пакетов:")
for pkg in packages
    try
```

Проверка пакетов

Запускаю скрипт. Все пакеты проверены. Можно приступать к работе с ними.

```
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project> julia --project=. scripts/test_script.jl
✓ Проект активирован: C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project

Проверка пакетов:
✓ DrWatson
✓ DifferentialEquations
✓ Plots
✓ DataFrames
✓ CSV
✓ JLD2
✓ Literate
✓ IJulia
✓ BenchmarkTools
✓ Quarto

Структура проекта:
Корень:      C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project
Данные:      C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project\data
Скрипты:    C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project\src
Графики:    C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project\plots
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project>
```

Рисунок 16: Пакеты успешно проверены

Работа с моделью экспоненциального роста

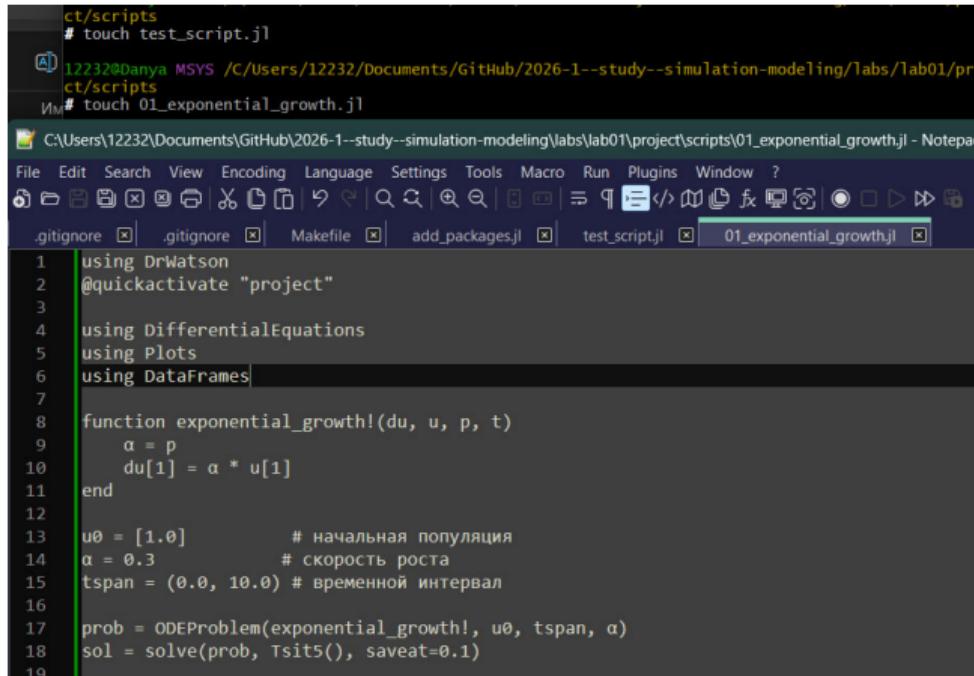
- Экспоненциальный рост — это процесс увеличения величины, при котором скорость роста в каждый момент времени пропорциональна текущему значению этой величины. Чем больше система, тем быстрее она растет.

Работа с моделью экспоненциального роста

- Экспоненциальный рост — это процесс увеличения величины, при котором скорость роста в каждый момент времени пропорциональна текущему значению этой величины. Чем больше система, тем быстрее она растет.
- В лабораторной работе требуется реализовать данную модель с помощью Julia.

Работа с моделью экспоненциального роста

Создаю скрипт 01_exponential_growth.jl, данный скрипт получит на вход один набор параметров для модели экспоненциального роста, получит решение, добавит их в таблицу, а также нарисует график.



```
ct/scripts
# touch test_script.jl
12232@Danya MSYS /C/Users/12232/Documents/GitHub/2026-1--study--simulation-modeling/labs/lab01/pr
ct/scripts
ИМ# touch 01_exponential_growth.jl
C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project\scripts\01_exponential_growth.jl - Notepad
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
.ignores .gitignore Makefile add_packages.jl test_script.jl 01_exponential_growth.jl
1 using DrWatson
2 @quickactivate "project"
3
4 using DifferentialEquations
5 using Plots
6 using DataFrames
7
8 function exponential_growth!(du, u, p, t)
9     α = p
10    du[1] = α * u[1]
11 end
12
13 u₀ = [1.0]          # начальная популяция
14 α = 0.3            # скорость роста
15 tspan = (0.0, 10.0) # временной интервал
16
17 prob = ODEProblem(exponential_growth!, u₀, tspan, α)
18 sol = solve(prob, Tsit5(), saveat=0.1)
```

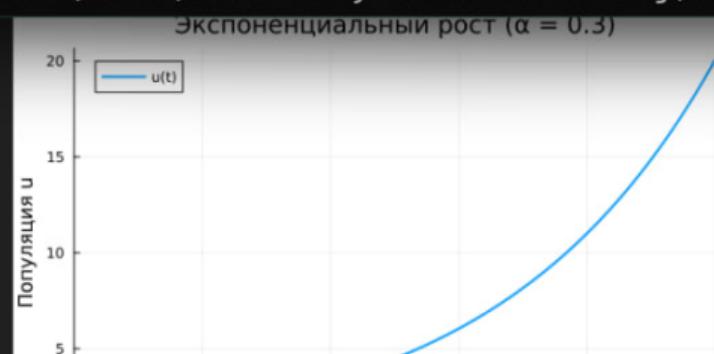
Результаты работы программы

Запускаю скрипт, работа выполнено успешно, получены результаты, а также график.

```
Графики:          C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project\plots
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project> julia --project=. scripts/01_ex.jl
Первые 5 строк результатов:
5x2 DataFrame
Row | t           u
     | Float64    Float64
---|-----
1  | 0.0         1.0
2  | 0.1         1.03045
3  | 0.2         1.06184
4  | 0.3         1.09417
5  | 0.4         1.1275
```

Аналитическое время удвоения: 2.31

```
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project>
```



Преобразование кода в литературный вид

Теперь требуется преобразовать программный код данного скрипта в литературный вид.

Литературное программирование

- Литературное (грамотное) программирование — это подход, приоритизирующий понятность программы для человека, а не её исполнение компьютером. В экосистеме Julia он реализуется через несколько инструментов.

Работа с кодом в литературном виде

Преобразую код 01_exponential_growth.jl в литературный вид.

The screenshot shows a Jupyter Notebook interface with three tabs: 'Makefile' (closed), 'add_packages.jl' (closed), and '01_exponential_growth.jl'. The code in the '01_exponential_growth.jl' tab is as follows:

```
1 # # Экспоненциальный рост
2 # **Цель:** Исследовать решение уравнения $du/dt = au$.
3 #
4 # ## Инициализация проекта и загрузка пакетов
5 using DrWatson
6 @quickactivate "project"
7
8 using DifferentialEquations
9 using Plots
10 using DataFrames
11 using JLD2
12
13 script_name = splitext(basename(PROGRAM_FILE))[1]
14 mkpath(plotsdir(script_name))
15 mkpath(datadir(script_name))
16
17 # ## Определение модели
18 # Уравнение экспоненциального роста:
19 # $\frac{du}{dt} = a u, \quad u(0) = u_0$.
20
21 function exponential_growth!(du, u, p, t)
22     a = p
23     du[1] = a * u[1]
24 end
25
26 # ## Первый запуск с параметрами по умолчанию
27 # Зададим начальные параметры:
28 u0 = [1.0]          # начальная популяция
29 a = 0.3            # скорость роста
30 tspan = (0.0, 10.0) # временной интервал
31
32 prob = ODEProblem(exponential_growth!, u0, tspan, a)
33 sol = solve(prob, Tsit5(), saveat=0.1)
```

The notebook also includes standard navigation icons for back, forward, and search at the bottom.

Работа с кодом в литературном виде

Т. к. сама суть исполняемого кода не была изменена, результат выполнения программы такой же. Рисунок также был создан.

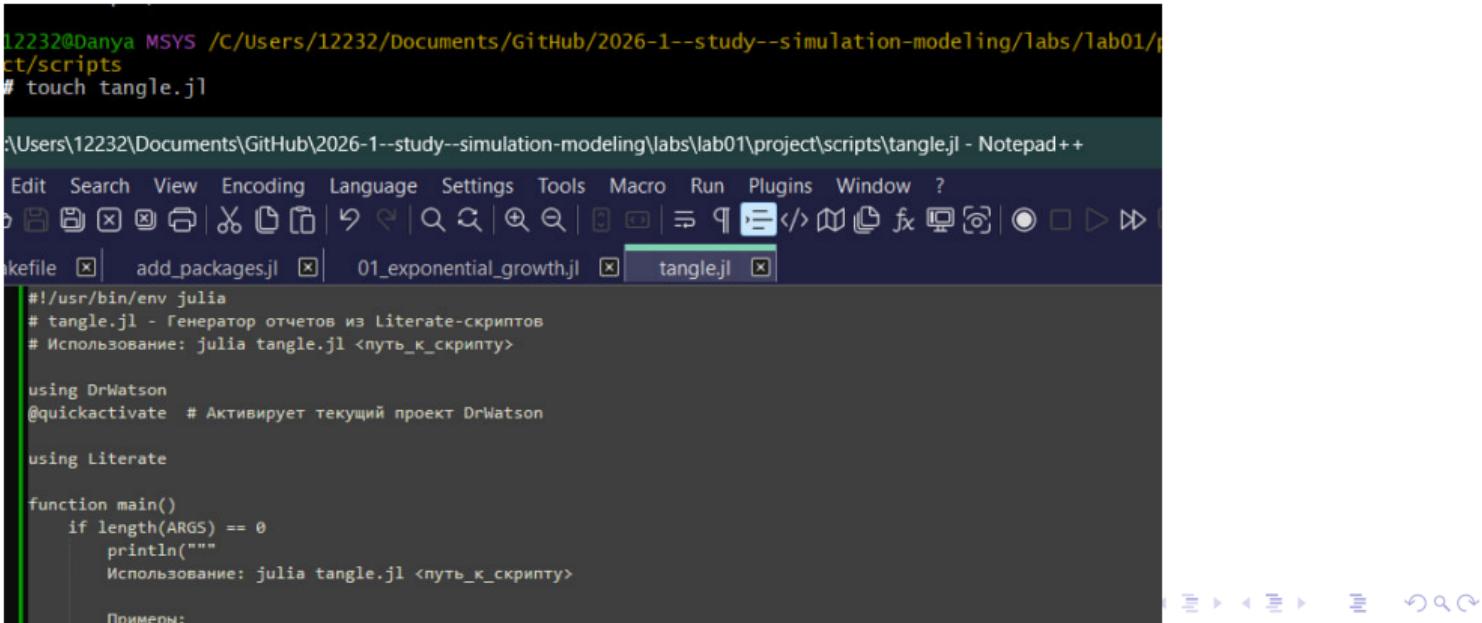
```
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project> julia --project=. scripts/01_exponential_growth.jl
Первые 5 строк результатов:
5x2 DataFrame
Row | t      u
     | Float64 Float64
1   | 0.0    1.0
2   | 0.1    1.03045
3   | 0.2    1.06184
4   | 0.3    1.09417
5   | 0.4    1.1275

Аналитическое время удвоения: 2.31
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project>
```

Рисунок 20: Результат 01_exponential_growth.jl в литературном виде

Формирование производных 01_exponential_growth

Теперь можно воспользоваться особенностями такого вида программирования. По заданию требуется создать скрипт `tangle.jl`, который на вход будет получать код в литературном виде, а на выходе будет производить три файла - чистый код, `jupyter notebook`, документацию в формате `Quarto`.



The screenshot shows a terminal session and a Notepad++ window side-by-side.

In the terminal (left side), the user is in their GitHub repository for '2026-1--study--simulation-modeling' and has run the command `# touch tangle.jl`. The path shown is `/C/Users/12232/Documents/GitHub/2026-1--study--simulation-modeling/labs/lab01/project/scripts`.

In the Notepad++ window (right side), the file `tangle.jl` is open. It contains the following Julia code:

```
#!/usr/bin/env julia
# tangle.jl - Генератор отчетов из Literate-скриптов
# Использование: julia tangle.jl <путь_к_скрипту>

using DrWatson
@quickactivate # Активирует текущий проект DrWatson

using Literate

function main()
    if length(ARGS) == 0
        println("""
        Использование: julia tangle.jl <путь_к_скрипту>
        """)

    Примеды:
```

Формирование производных 01_exponential_growth

Запускаю tangle.jl, все три файла успешно созданы.

```
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project> julia --project=. scripts/tangle.jl scripts/01_exponential_growth
`генерация из: scripts/01_exponential_growth.jl
Info: generating plain script file from `C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project\scripts\01_exponential_gro
jl`
Info: writing result to `C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project\scripts\01_exponential_growth\01_exponenti
al_growth.jl`
✓ Чистый скрипт: C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project\scripts\01_exponential_growth\01_exponenti
al_growth.jl
Info: generating markdown page from `C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project\scripts\01_exponential_
growth.qmd`
Info: writing result to `C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project\markdown\01_exponential_growth\01_exponent
ial_growth.qmd`
✓ Quarto: C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project\markdown\01_exponential_growth\01_exponential_growth.qmd
Info: generating notebook from `C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project\scripts\01_exponential_growth.jl`
Info: writing result to `C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project\notebooks\01_exponential_growth\01_exponen
tial.ipynb`
✓ Notebook: C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project\notebooks\01_exponential_growth\01_exponential_growth.i
nbs
Готово! Все файлы созданы.
PS C:\Users\12232\Documents\GitHub\2026-1--study--simulation-modeling\labs\lab01\project>
```

Рисунок 22: Результат работы tangle.jl

Формирование производных 01_exponential_growth

Проверим созданный Jupyter notebook. Файл открывается и успешно работает.

Экспоненциальный рост

Цель: Исследовать решение уравнения $du/dt = \alpha u$.

Инициализация проекта и загрузка пакетов

```
In [ ]: using DrWatson  
@quickactivate "project"  
  
using DifferentialEquations  
using Plots  
using DataFrames  
using JLD2  
  
script_name = splitext(basename(PROGRAM_FILE))[1]  
mkpath(plotstadir(script_name))  
mkpath(datadir(script_name))
```

Определение модели

Уравнение экспоненциального роста: $\frac{du}{dt} = \alpha u$, $u(0) = u_0$

```
In [ ]: function exponential_growth!(du, u, p, t)  
    α = p  
    du[1] = α * u[1]  
end
```

Первый запуск с параметрами по умолчанию

Зададим начальные параметры:



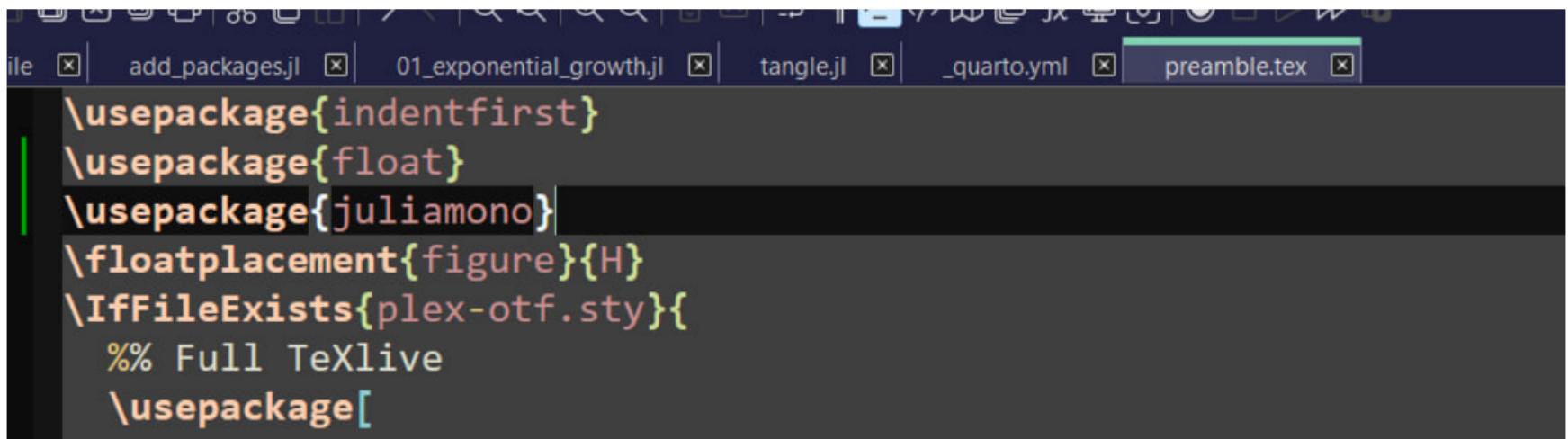
Интеграция 01_exponential_growth в отчёт

Теперь требуется внести код 01_exponential_growth в отчёт. Перед этим в файл _quarto.yml добавляю код, включающий поддержку Julia.

```
решение |-- add_packages.jl |-- 01_exponential_growth.jl |-- tangie.jl |-- _quarto.yml
└── _quarto.yml
    ├── project:
    │   ├── title: "report"
    │   ├── output-dir: "_output"
    │
    │   ├── standalone: true
    │   ├── self-contained: true
    │
    │   ## Julia support
    │   ├── engine: julia
    │   └── julia:
```

Интеграция 01_exponential_growth в отчёт

Затем вношу изменения в preamble.tex, также добавляя поддержку Julia.



```
\usepackage{indentfirst}
\usepackage{float}
\usepackage{juliamono}
\floatplacement{figure}{H}
\IfFileExists{plex-otf.sty}{
    %% Full TeXlive
    \usepackage[
```

Рисунок 25: Изменения в preamble.tex

Интеграция 01_exponential_growth в отчёт

Код успешно добавлен.

5. Отчёт 01_exponential_growth

6. Экспоненциальный рост

Цель: Исследовать решение уравнения $du/dt = \alpha u$.

6.1 Инициализация проекта и загрузка пакетов

```
using DrWatson
@quickactivate "../project"

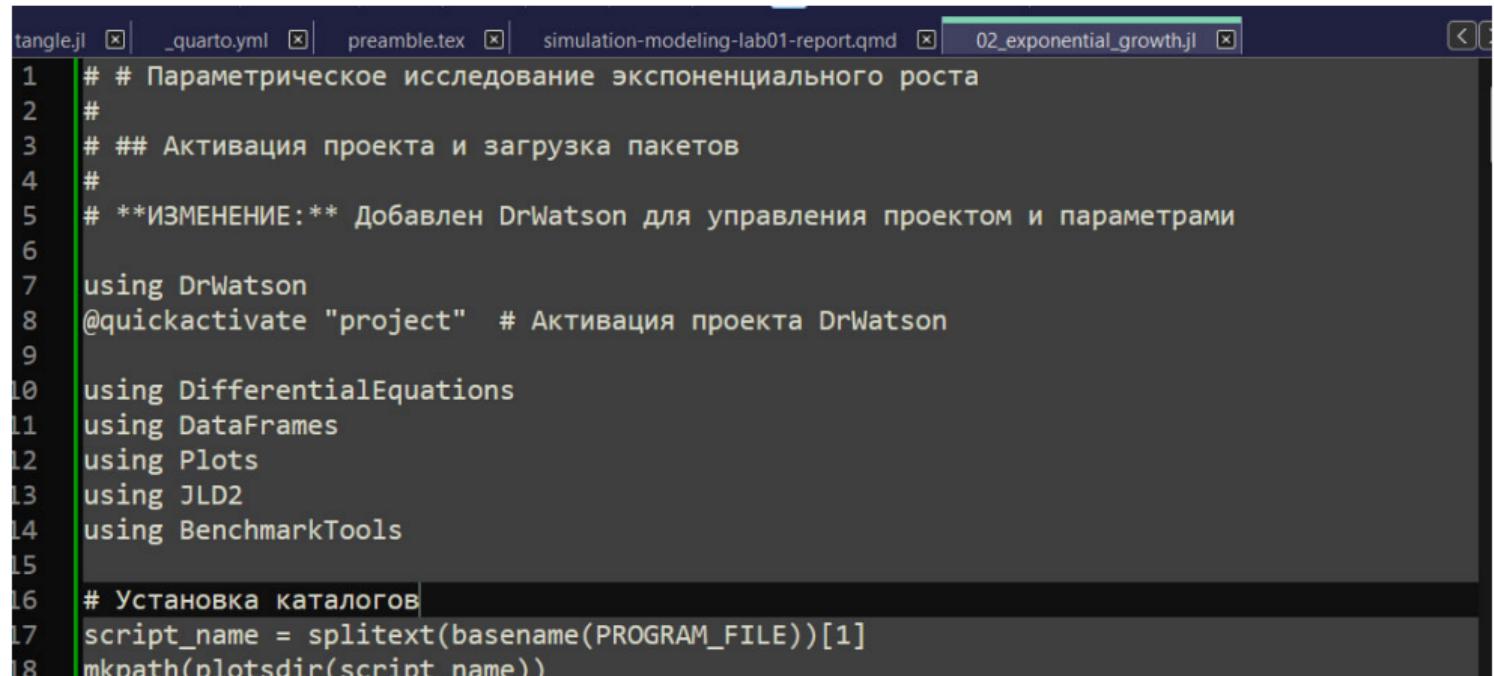
using Plots
default(fmt = :png)
gr(fmt = :png)

using DifferentialEquations
using DataFrames
```



Реализация модели с параметрами

Теперь требуется создать новый код, вместо того, чтобы подавать один набор данных, как в 01_exponential_growth.jl, изменю программу так, чтобы подавалось несколько наборов данных. Код уже преобразован в литературный вид.



```
1 # # Параметрическое исследование экспоненциального роста
2 #
3 # ## Активация проекта и загрузка пакетов
4 #
5 # **ИЗМЕНЕНИЕ:** Добавлен DrWatson для управления проектом и параметрами
6
7 using DrWatson
8 @quickactivate "project" # Активация проекта DrWatson
9
10 using DifferentialEquations
11 using DataFrames
12 using Plots
13 using JLD2
14 using BenchmarkTools
15
16 # Установка каталогов
17 script_name = splitext(basename(PROGRAM_FILE))[1]
18 mkpath(plotspath(script_name))
```

Реализация модели с параметрами

Запускаю программу, код работает успешн.

```
Среднее время: 0.0 сек
енчмарк для а = 0.3:
Среднее время: 0.0 сек
енчмарк для а = 0.5:
Среднее время: 0.0 сек
енчмарк для а = 0.8:
Среднее время: 0.0001 сек
енчмарк для а = 1.0:
Среднее время: 0.0 сек
=====
АБОРАТОРНАЯ РАБОТА ЗАВЕРШЕНА
=====

результаты сохранены в:
• data/02_exponential_growth/single/           - базовый эксперимент
• data/02_exponential_growth/parametric_scan/   - параметрическое сканирование
• data/02_exponential_growth/all_results.jld2    - сводные данные
• plots/02_exponential_growth/                  - все графики
• data/02_exponential_growth/all_plots.jld2     - объекты графиков
```



Реализация модели с параметрами

Программой tangle.jl создаю три файла из 02_exponential_growth.jl, среди которых Jupyter notebook. Он успешно работает.

Параметрическое исследование экспоненциального роста

Активация проекта и загрузка пакетов

ИЗМЕНЕНИЕ: Добавлен DrWatson для управления проектом и параметрами

```
In [ ]: using DrWatson  
@quickactivate "project" # Активация проекта DrWatson  
  
using DifferentialEquations  
using DataFrames  
using Plots  
using JLD2  
using BenchmarkTools
```

Установка каталогов

```
In [ ]: script_name = splitext(basename(PROGRAM_FILE))[1]  
mkpath(plotspath(script_name))  
mkpath(datadir(script_name))
```

Определение модели

Модель: $du/dt = \alpha \cdot u$

```
In [ ]: function exponential_growth!(du, u, p, t)  
    α = p.α # *ИЗМЕНЕНИЕ:/* Параметры теперь передаются как именованный кортеж  
    du[1] = α * u[1]  
end
```



Реализация модели с параметрами

Добавлю эту программу в отчёт.

```
59
60 # Отчёт 01_exponential_growth
61
62 {{< include ../project/markdown/01_exponential_growth/01_exponential_growth.qmd >}}
63
64 # Отчёт 02_exponential_growth
65
66 {{< include ../project/markdown/02_exponential_growth/02_exponential_growth.qmd >}}
67
68 # Выводы
```

Рисунок 30: 02_exponential_growth.jl внутри отчёта

Реализация модели с параметрами

Код успешно интегрирован в отчёт, его вы можете просмотреть ниже:
экспоненциального роста

8.1 Активация проекта и загрузка пакетов

ИЗМЕНЕНИЕ: Добавлен DrWatson для управления проектом и параметрами

```
using DrWatson
@quickactivate "../project" # активирует проект DrWatson

using Plots
default(fmt = :png)
gr(fmt = :png)

using DifferentialEquations
using DataFrames
using JLD2
```

Раздел 2

Выводы

Выводы

В результате выполнения данной лабораторной работы, я вспомнил основные методы работы с системой Git и разными платформами, а также получил практические навыки работы с Julia.

Раздел 3

Список литературы

Список литературы

- Лабораторная работа №1