

Quant Modelling

Backtest investment strategies, 1990-2023

Rank	Model Name	Author	Language	Files	Sharpe Ratio	Annual Return	Volatility	Max Drawdown	Year
1	Implementation of the Nonlinear Portfolio Transformer Submitted Apr 3, 2026 · Scored Apr 4, 2026	Gareth Campbell	Python	 	2.7879	27.88%	10.00%	-18.74%	2024
2	LightGBM + Investment Families Submitted Apr 22, 2026 · Scored Apr 28, 2026	Darren Li	Python		2.1854	21.85%	10.00%	-19.40%	2024
3	MAXSER Submitted Jan 13, 2026 · Scored Jan 13, 2026	Yingying Li	R	 	2.0261	20.26%	10.00%	-13.23%	2024
4	Benchmark: Instrumented PCA factor portfolio  Benchmark	—	Python		1.9481	19.48%	10.00%	-9.92%	2024
5	ClusterSharpe Selection + MeanVar Weighting + Linear LS Submitted Mar 10, 2026 · Scored Mar 11, 2026	Gareth Campbell	Python		1.5010	15.01%	10.00%	-20.71%	2024
6	Benchmark: Kozak-Nagel-Santosh rank-weighted factors portfolio  Benchmark	—	R		1.4195	14.20%	10.00%	-8.12%	2024
7	Bull/Bear Naive Submitted Feb 20, 2026 · Scored Feb 24, 2026	Kavita Kar	Python	 	1.3762	13.76%	10.00%	-23.86%	2024
8	Cross-Sectional Return Ensemble (CSRE) Submitted Feb 21, 2026 · Scored Feb 23, 2026	Anna Siamionava	Python	 	1.2591	12.59%	10.00%	-29.00%	2024
9	Rolling Factor Selection with RankSharpe Weighting Submitted Mar 9, 2026 · Scored Mar 10, 2026	Gareth Campbell	Python		1.1895	11.89%	10.00%	-15.19%	2024
10	Grid-Optimized Group Lasso (GOGL) Submitted Feb 19, 2026 · Scored Feb 20, 2026	mbn	Python		1.1030	11.03%	10.00%	-29.37%	2024

VS Code and Claude Code

- Ensure that you have installed VS Code, and added the Claude Code extension
- Follow the instructions from previous slides to ensure you have access

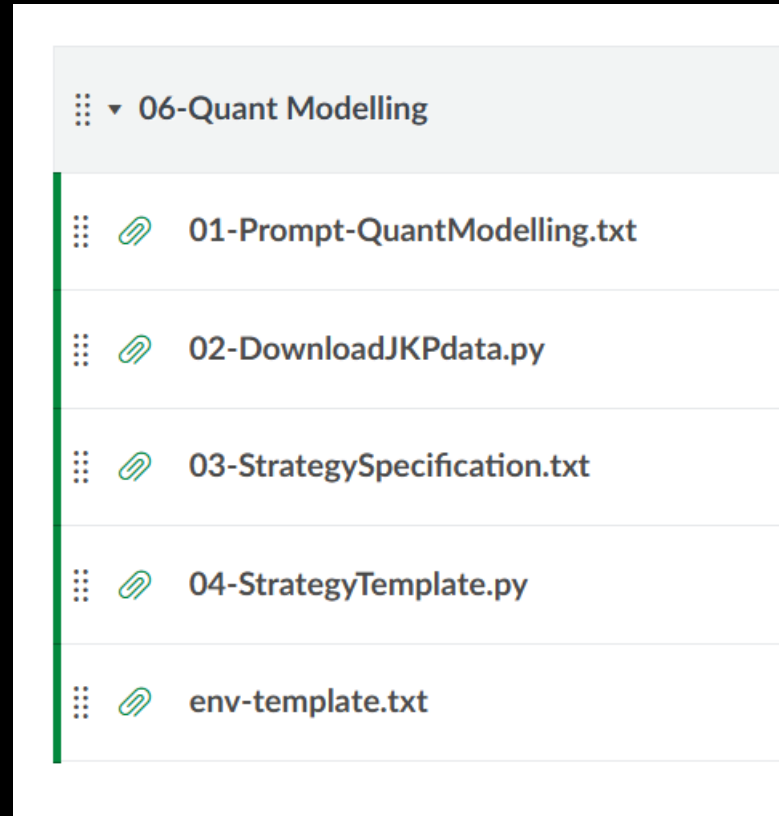
Python

- For this task you also need to have Python installed on your PC
- Download from <https://www.python.org>

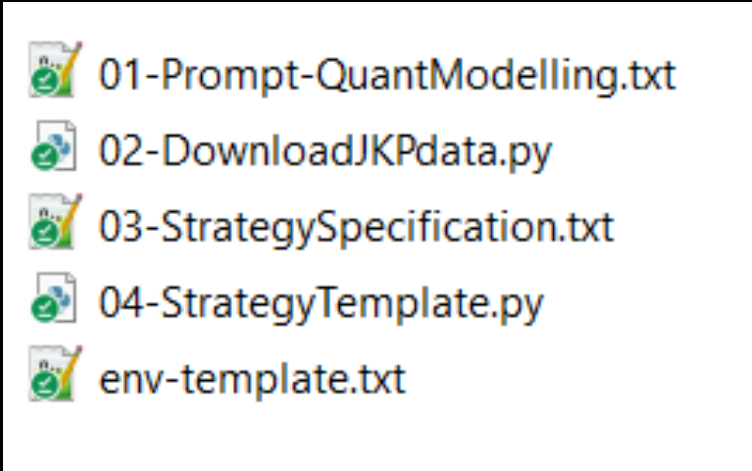
WRDS

- We will access the data through WRDS
- Register for an account at:
- <https://wrds-www.wharton.upenn.edu/>
- Copy your username and password to the env-example.txt file

Download the files from the 06-Quant Modelling section on Canvas



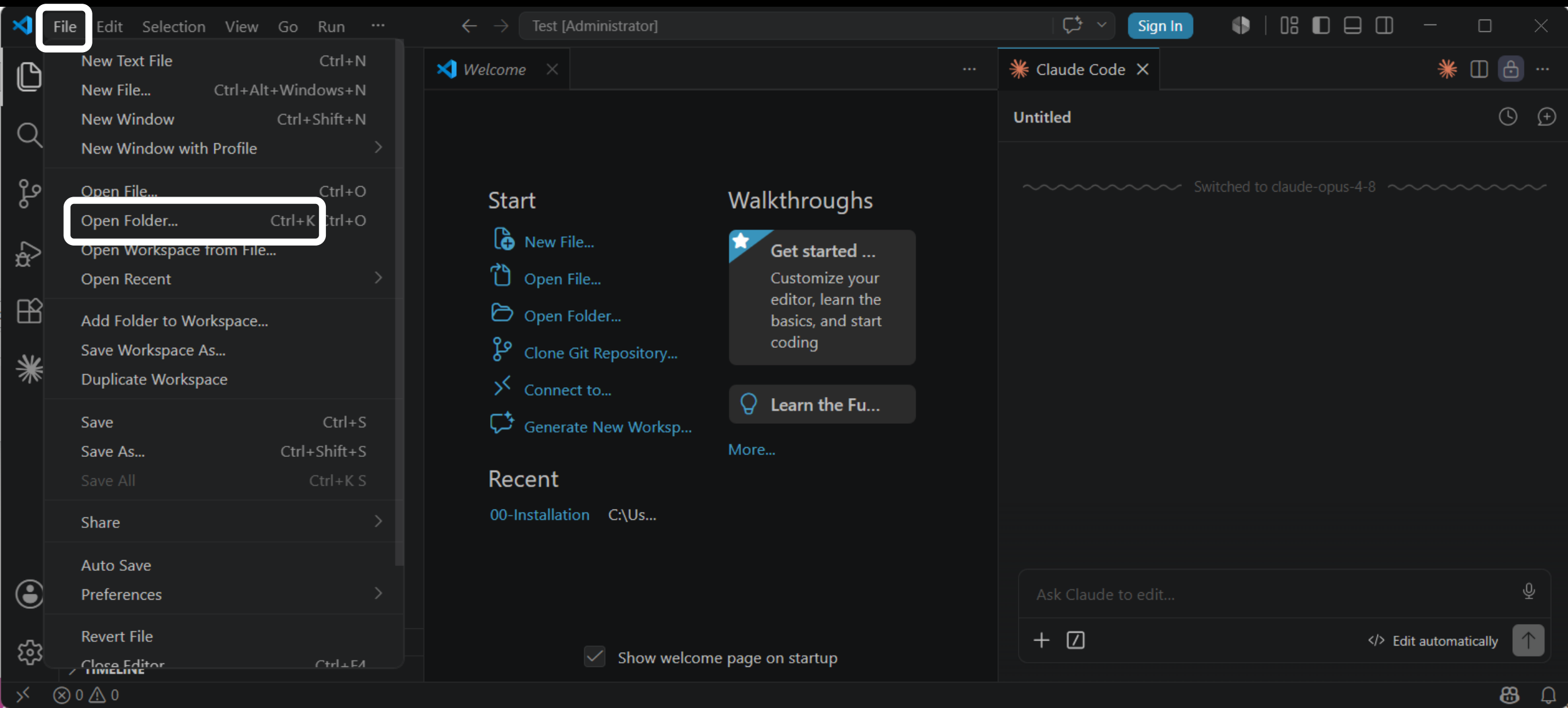
Save the files to a folder on your PC



A screenshot of a file explorer window with a white background. It displays five files in a list. Each file has a small icon to its left: a document with a green checkmark and a yellow pencil for the .txt files, and a document with a green checkmark and a blue circular arrow for the .py files. The files are listed as follows:

- 01-Prompt-QuantModelling.txt
- 02-DownloadJKPdata.py
- 03-StrategySpecification.txt
- 04-StrategyTemplate.py
- env-template.txt

In VS Code, click File -> Open Folder Select the folder containing those files



Click on env-template.txt

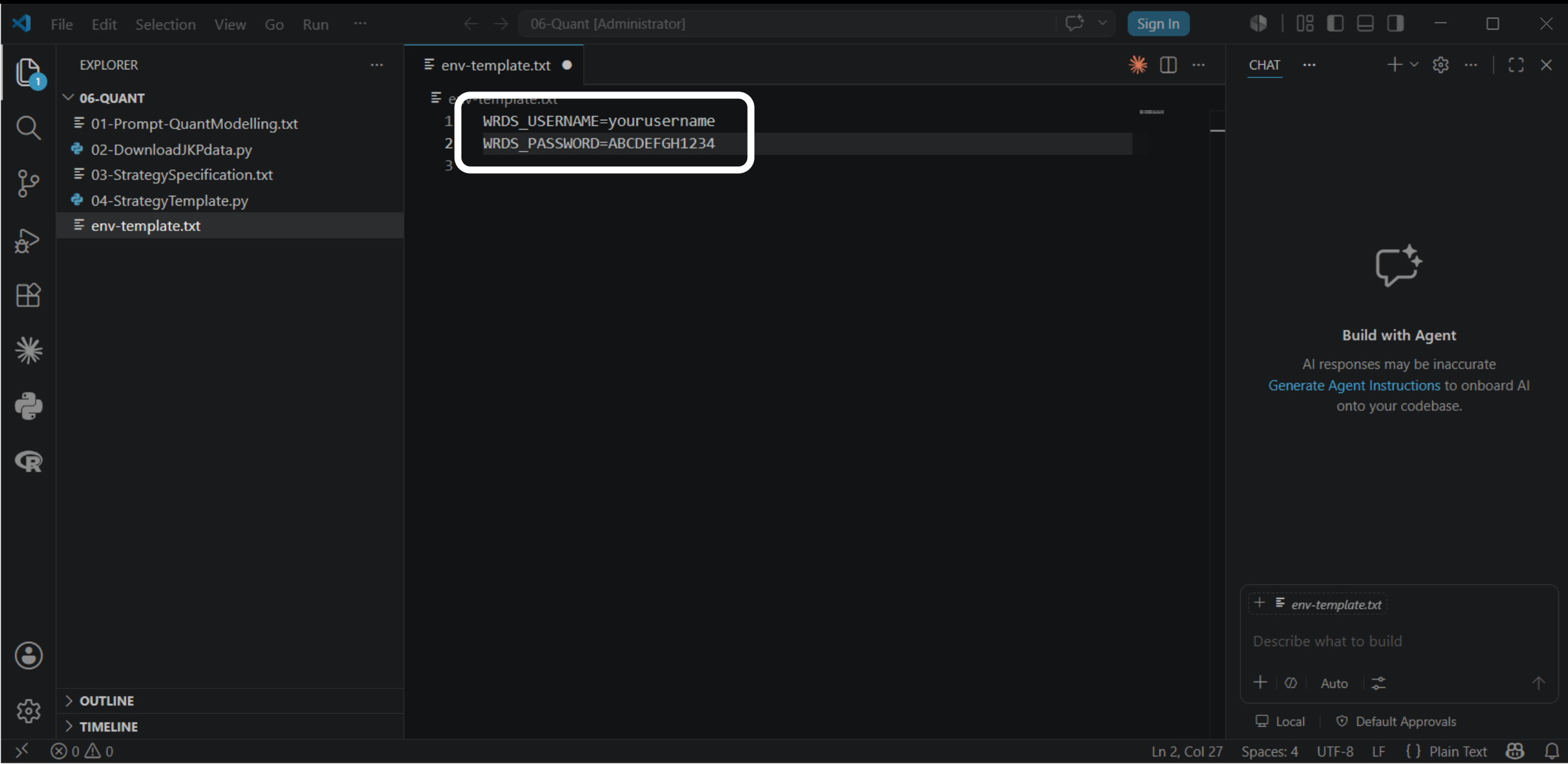
The screenshot shows the Visual Studio Code interface with the following components:

- Explorer Sidebar:** Displays the file structure under the folder '06-QUANT'. The files listed are '01-Prompt-QuantModelling.txt', '02-DownloadJKPdata.py', '03-StrategySpecification.txt', '04-StrategyTemplate.py', and 'env-template.txt'. The file 'env-template.txt' is highlighted with a white border.
- Main Editor:** Shows the content of 'env-template.txt' with the following text:

```
1 WRDS_USERNAME=  
2 WRDS_PASSWORD=  
3
```
- Chat Sidebar:** Displays a chat interface with the heading 'Build with Agent' and the text 'AI responses may be inaccurate. Generate Agent Instructions to onboard AI onto your codebase.' Below this, there is a text input field with a placeholder 'Describe what to build' and a button 'Generate Agent Instructions'. At the bottom, there are options for 'Local' and 'Default Approvals'.

The status bar at the bottom indicates the current position is 'Ln 1, Col 1' and the file encoding is 'UTF-8'.

Add your username and password



The screenshot shows the Visual Studio Code interface with the Explorer sidebar on the left. The file explorer shows a folder named '06-QUANT' containing several files: '01-Prompt-QuantModelling.txt', '02-DownloadJKPdata.py', '03-StrategySpecification.txt', '04-StrategyTemplate.py', and 'env-template.txt'. The 'env-template.txt' file is selected and its content is displayed in the editor. The content of the file is:

```
1 WRDS_USERNAME=yourusername
2 WRDS_PASSWORD=ABCDEFGH1234
3
```

A white rectangular box highlights the two lines of code. The status bar at the bottom indicates 'Ln 2, Col 27', 'Spaces: 4', 'UTF-8', 'LF', and 'Plain Text'. On the right side, there is a 'CHAT' panel with a 'Build with Agent' section. The text in this section reads: 'AI responses may be inaccurate. Generate Agent Instructions to onboard AI onto your codebase.' Below this, there is a text input field with a placeholder 'Describe what to build' and a 'Send' button. At the bottom right, there are icons for 'Local' and 'Default Approvals'.

Click on 01-Prompt-QuantModelling.txt

The image shows a Visual Studio Code (VS Code) interface. The Explorer sidebar on the left displays a file tree for a project named '06-QUANT'. The file '01-Prompt-QuantModelling.txt' is selected and highlighted with a white rectangle. The main editor area shows the content of this file, which is a text document containing instructions for setting up a quant modeling environment. The Chat sidebar on the right is open, showing a 'Build with Agent' section with a warning that AI responses may be inaccurate and a button to 'Generate Agent Instructions'. Below this, there is a text input field for describing what to build, and a 'Local' button for default approvals.

File Edit Selection View Go Run ... 06-Quant [Administrator] Sign In

EXPLORER

- 06-QUANT
 - 01-Prompt-QuantModelling.txt
 - 02-DownloadJKPdata.py
 - 03-StrategySpecification.txt
 - 04-StrategyTemplate.py
 - env-template.txt

01-Prompt-QuantModelling.txt

```
1 =====
2 REQUIREMENTS - Please ensure before running:
3 =====
4 1. PYTHON: Install Python 3.x from https://www.python.org/downloads/
5 | - Check "Add Python to PATH" during installation
6
7 2. PACKAGES: Run in terminal:
8 | pip install numpy pandas scipy pyarrow wrds python-dotenv
9
10 3. DATA: The strategy needs three parquet files:
11 | ctff_chars.parquet, ctff_features.parquet, ctff_daily_ret.parquet (optional)
12 | - Either use the course-provided files, OR
13 | - Build them yourself from WRDS (requires a WRDS account) using
14 | 02-DownloadJKPdata.py - see its header for full instructions.
15
16 4. WRDS CREDENTIALS: You will need a subscription to CRSP/Compustat via WRDS, which
17 =====
18
19 SETUP STEP (do this before anything else)
20 Ask the user for the folder containing their data files, and check that
21 ctff_chars.parquet and ctff_features.parquet exist there.
22 If they DO exist, skip straight to the tasks below.
23 If they do NOT exist, offer to build the data with 02-DownloadJKPdata.py:
24 | - Check whether a .env file exists in the root folder. If it does not,
25 | create one by copying env-template.txt to a file named .env in the root
26 | folder. Then ask the user for their WRDS username and password, and
27 | replace your_username_here / your_password_here in .env with the values
28 | they provide. (The user needs their own WRDS account.)
29 | - Then run: python 02-DownloadJKPdata.py --out-dir <folder>
```

CHAT

Build with Agent

AI responses may be inaccurate

Generate Agent Instructions to onboard AI onto your codebase.

+ 01-Prompt-QuantModelling.txt

Describe what to build

+ | | Auto | |

Local | Default Approvals

Ln 1, Col 1 Spaces: 3 UTF-8 LF { } Plain Text

Read through the prompt

The image shows a Visual Studio Code editor window with the file `01-Prompt-QuantModelling.txt` open. A white oval highlights the first four numbered instructions in the file:

1. PYTHON: Install Python 3.x from <https://www.python.org/downloads/>
 - Check "Add Python to PATH" during installation
2. PACKAGES: Run in terminal:
`pip install numpy pandas scipy pyarrow wrds python-dotenv`
3. DATA: The strategy needs three parquet files:
`ctff_chars.parquet`, `ctff_features.parquet`, `ctff_daily_ret.parquet` (optional)
 - Either use the course-provided files, OR
 - Build them yourself from WRDS (requires a WRDS account) using `02-DownloadJKPdata.py` - see its header for full instructions.
4. WRDS CREDENTIALS: You will need a subscription to CRSP/Compustat via WRDS, which

The right sidebar shows a chat interface with a section titled "Build with Agent". Below this, it says "AI responses may be inaccurate" and "Generate Agent Instructions to onboard AI onto your codebase." At the bottom of the sidebar, there is a text input field labeled "Describe what to build" and a button labeled "Auto".

Explanation of prompt

- Begins with a summary of installation requirements
- Downloads the data from WRDS
- Sorts stocks by characteristics, and evaluates which characteristics were previously associated with high performance
- Weights characteristics and develops portfolios
- Evaluates the performance of this strategy from 1990-2023
- Provides the outputs required for submission at:
<https://jkpfactors.com/ctf/submit>

Click orange icon to open Claude Code

The screenshot displays the Visual Studio Code editor interface. The top menu bar includes File, Edit, Selection, View, Go, Run, and a 'Sign In' button. The Explorer sidebar on the left shows a project named '06-QUANT' with files: 01-Prompt-QuantModelling.txt, 02-DownloadJKPdata.py, 03-StrategySpecification.txt, 04-StrategyTemplate.py, and env-template.txt. The main editor window is open to '01-Prompt-QuantModelling.txt', which contains instructions for setting up a Python environment and downloading data. An orange star icon, representing the Claude Code extension, is highlighted with a white box in the top right of the editor area. The right sidebar shows the 'CHAT' panel with a 'Build with Agent' section and a text input field for describing what to build.

```
01-Prompt-QuantModelling.txt
1  =====
2  REQUIREMENTS - Please ensure before running:
3  =====
4  1. PYTHON: Install Python 3.x from https://www.python.org/downloads/
5  |   - Check "Add Python to PATH" during installation
6
7  2. PACKAGES: Run in terminal:
8  |   pip install numpy pandas scipy pyarrow wrds python-dotenv
9
10 3. DATA: The strategy needs three parquet files:
11 |   ctff_chars.parquet, ctff_features.parquet, ctff_daily_ret.parquet (optional)
12 |   - Either use the course-provided files, OR
13 |   - Build them yourself from WRDS (requires a WRDS account) using
14 |   02-DownloadJKPdata.py - see its header for full instructions.
15
16 4. WRDS CREDENTIALS: You will need a subscription to CRSP/Compustat via WRDS, which
17  =====
18
19 SETUP STEP (do this before anything else)
20 Ask the user for the folder containing their data files, and check that
21 ctff_chars.parquet and ctff_features.parquet exist there.
22 If they DO exist, skip straight to the tasks below.
23 If they do NOT exist, offer to build the data with 02-DownloadJKPdata.py:
24 |   - Check whether a .env file exists in the root folder. If it does not,
25 |   create one by copying env-template.txt to a file named .env in the root
26 |   folder. Then ask the user for their WRDS username and password, and
27 |   replace your_username_here / your_password_here in .env with the values
28 |   they provide. (The user needs their own WRDS account.)
29 |   - Then run: python 02-DownloadJKPdata.py --out-dir <folder>
```

Close the other chat window

The screenshot displays the Claude Code IDE interface. The main editor window shows a file named `01-Prompt-QuantModelling.txt` with the following content:

```
1 =====
2 REQUIREMENTS - Please ensure before
3 =====
4 1. PYTHON: Install Python 3.x from h
5 | - Check "Add Python to PATH" duri
6
7 2. PACKAGES: Run in terminal:
8 | pip install numpy pandas scipy py
9
10 3. DATA: The strategy needs three parquet
11 | ctff_chars.parquet, ctff_features.parc
12 | - Either use the course-provided files
13 | - Build them yourself from WRDS (requi
14 | 02-DownloadJKPdata.py - see its head
15
16 4. WRDS CREDENTIALS: You will need a subs
17 =====
18
19 SETUP STEP (do this before anything else)
20 Ask the user for the folder containing th
21 ctff_chars.parquet and ctff_features.parc
22 If they DO exist, skip straight to the ta
23 If they do NOT exist, offer to build the
24 | - Check whether a .env file exists in t
25 | create one by copying env-template.t
26 | folder. Then ask the user for their v
27 | replace your_username_here / your_pas
28 | they provide. (The user needs their c
29 | - Then run: python 02-DownloadJKPdata.py
```

The right sidebar contains the **CHAT** panel, which is currently closed. The **CHAT** button is highlighted with a white box. The chat panel shows the **Build with Agent** section, which includes the text: "AI responses may be inaccurate. Generate Agent Instructions to onboard AI onto your codebase." Below this, there is a text input field labeled "Describe what to build" and a button labeled "Auto".

The bottom status bar shows the following information: `< 0 0 0` (on the left) and `Local | Default Approvals` (on the right).

Ensure Settings use Model=Opus, Effort=High, and Auto Mode

The screenshot shows the Visual Studio Code interface with the Claude Code extension. The Explorer panel on the left shows a project named '06-QUANT' with files: '01-Prompt-QuantModelling.txt', '02-DownloadJKPdata.py', '03-StrategySpecification.txt', '04-StrategyTemplate.py', and 'env-template.txt'. The main editor displays the content of '01-Prompt-QuantModelling.txt', which contains instructions for setting up a Python environment and downloading data. The Claude Code sidebar on the right shows the 'Untitled' document and a notification that 'Auto mode is enabled'. At the bottom of the sidebar, there are two buttons: a square icon with a plus sign and a button labeled 'Auto mode' with a lightning bolt icon. Both buttons are highlighted with white boxes.

File Edit Selection View Go Run ... 06-Quant [Administrator] Sign In

EXPLORER

- 06-QUANT
 - 01-Prompt-QuantModelling.txt
 - 02-DownloadJKPdata.py
 - 03-StrategySpecification.txt
 - 04-StrategyTemplate.py
 - env-template.txt

01-Prompt-QuantModelling.txt

```
1 =====
2 REQUIREMENTS - Please ensure before running:
3 =====
4 1. PYTHON: Install Python 3.x from https://www.python.org/
5 | - Check "Add Python to PATH" during installation
6
7 2. PACKAGES: Run in terminal:
8 | pip install numpy pandas scipy pyarrow wrds python-dotenv
9
10 3. DATA: The strategy needs three parquet files:
11 | ctff_chars.parquet, ctff_features.parquet, ctff_daily_ret.parquet
12 | - Either use the course-provided files, OR
13 | - Build them yourself from WRDS (requires a WRDS account) using
14 | 02-DownloadJKPdata.py - see its header for full instructions
15
16 4. WRDS CREDENTIALS: You will need a subscription to CRSP/Compustat
17 =====
18
19 SETUP STEP (do this before anything else)
20 Ask the user for the folder containing their data files, and check
21 ctff_chars.parquet and ctff_features.parquet exist there.
22 If they DO exist, skip straight to the tasks below.
23 If they do NOT exist, offer to build the data with 02-DownloadJKPdata.py
24 | - Check whether a .env file exists in the root folder. If it does
25 | create one by copying env-template.txt to a file named .env in the
26 | folder. Then ask the user for their WRDS username and password
```

Claude Code

Untitled

// TODO: Everything. Let's start.

Auto mode is enabled

Auto mode lets Claude handle permission prompts automatically. Claude checks each tool call for risky actions and prompt injection before executing, runs the ones it assesses as lower-risk, and blocks the rest.

Prefer the Terminal experience? Switch back in Settings.

ctrl esc to focus or unfocus Claude

01-Prompt-QuantModelling.txt

Auto mode

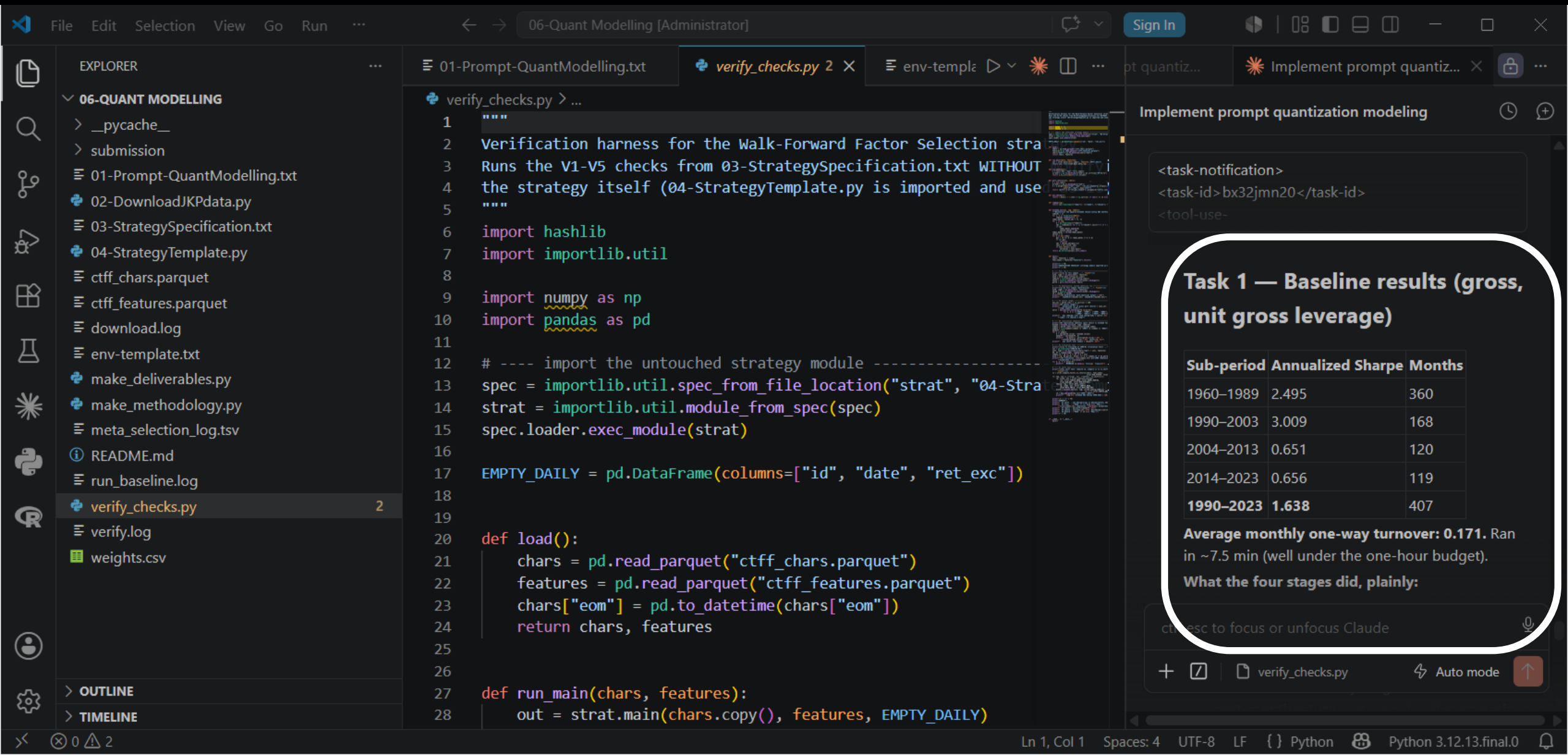
Ask Claude Code: implement 01-Prompt-QuantModelling.txt

The image shows a Visual Studio Code editor window with the following components:

- EXPLORER:** A file tree on the left showing a project named "06-QUANT" with files: "01-Prompt-QuantModelling.txt", "02-DownloadJKPdata.py", "03-StrategySpecification.txt", "04-StrategyTemplate.py", and "env-template.txt".
- Editor:** The main workspace displays the content of "01-Prompt-QuantModelling.txt". The text includes instructions for setting up a Python environment, installing packages (numpy, pandas, scipy, pyarrow, wrds, python-dotenv), and downloading data files. It also mentions the need for WRDS credentials and a setup step to ask the user for folder information.
- Claude Code Sidebar:** On the right, the Claude Code sidebar is open. It shows the Claude Code logo and the text "// TODO: Everything. Let's start." Below this, a notification states "Auto mode is enabled". At the bottom, there is a search bar containing the text "implement 01-Prompt-QuantModelling.txt".

```
1 =====
2 REQUIREMENTS - Please ensure before running:
3 =====
4 1. PYTHON: Install Python 3.x from https://www.python.org/
5 | - Check "Add Python to PATH" during installation
6
7 2. PACKAGES: Run in terminal:
8 | pip install numpy pandas scipy pyarrow wrds python-dotenv
9
10 3. DATA: The strategy needs three parquet files:
11 | ctff_chars.parquet, ctff_features.parquet, ctff_daily_ret.parquet
12 | - Either use the course-provided files, OR
13 | - Build them yourself from WRDS (requires a WRDS account) using
14 | 02-DownloadJKPdata.py - see its header for full instructions
15
16 4. WRDS CREDENTIALS: You will need a subscription to CRSP/Compustat
17 =====
18
19 SETUP STEP (do this before anything else)
20 Ask the user for the folder containing their data files, and check
21 ctff_chars.parquet and ctff_features.parquet exist there.
22 If they DO exist, skip straight to the tasks below.
23 If they do NOT exist, offer to build the data with 02-DownloadJKPdata.py
24 | - Check whether a .env file exists in the root folder. If it does
25 | create one by copying env-template.txt to a file named .env in the
26 | folder. Then ask the user for their WRDS username and password
```

It will backtest the strategy



06-Quant Modelling [Administrator]

EXPLORED

- 06-QUANT MODELLING
 - > __pycache__
 - > submission
 - 01-Prompt-QuantModelling.txt
 - 02-DownloadJKPdata.py
 - 03-StrategySpecification.txt
 - 04-StrategyTemplate.py
 - ctff_chars.parquet
 - ctff_features.parquet
 - download.log
 - env-template.txt
 - make_deliverables.py
 - make_methodology.py
 - meta_selection_log.tsv
 - README.md
 - run_baseline.log
 - verify_checks.py** 2
 - verify.log
 - weights.csv

01-Prompt-QuantModelling.txt

verify_checks.py 2

```
1 """
2 Verification harness for the Walk-Forward Factor Selection stra
3 Runs the V1-V5 checks from 03-StrategySpecification.txt WITHOUT
4 the strategy itself (04-StrategyTemplate.py is imported and use
5 """
6 import hashlib
7 import importlib.util
8
9 import numpy as np
10 import pandas as pd
11
12 # ---- import the untouched strategy module -----
13 spec = importlib.util.spec_from_file_location("strat", "04-Strat
14 strat = importlib.util.module_from_spec(spec)
15 spec.loader.exec_module(strat)
16
17 EMPTY_DAILY = pd.DataFrame(columns=["id", "date", "ret_exc"])
18
19
20 def load():
21     chars = pd.read_parquet("ctff_chars.parquet")
22     features = pd.read_parquet("ctff_features.parquet")
23     chars["eom"] = pd.to_datetime(chars["eom"])
24     return chars, features
25
26
27 def run_main(chars, features):
28     out = strat.main(chars.copy(), features, EMPTY_DAILY)
```

Implement prompt quantization modeling

<task-notification>
<task-id> bx32jmn20 </task-id>
<tool-use>

Task 1 — Baseline results (gross, unit gross leverage)

Sub-period	Annualized Sharpe	Months
1960–1989	2.495	360
1990–2003	3.009	168
2004–2013	0.651	120
2014–2023	0.656	119
1990–2023	1.638	407

Average monthly one-way turnover: 0.171. Ran in ~7.5 min (well under the one-hour budget).

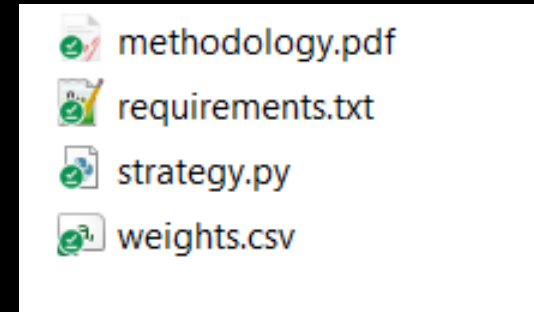
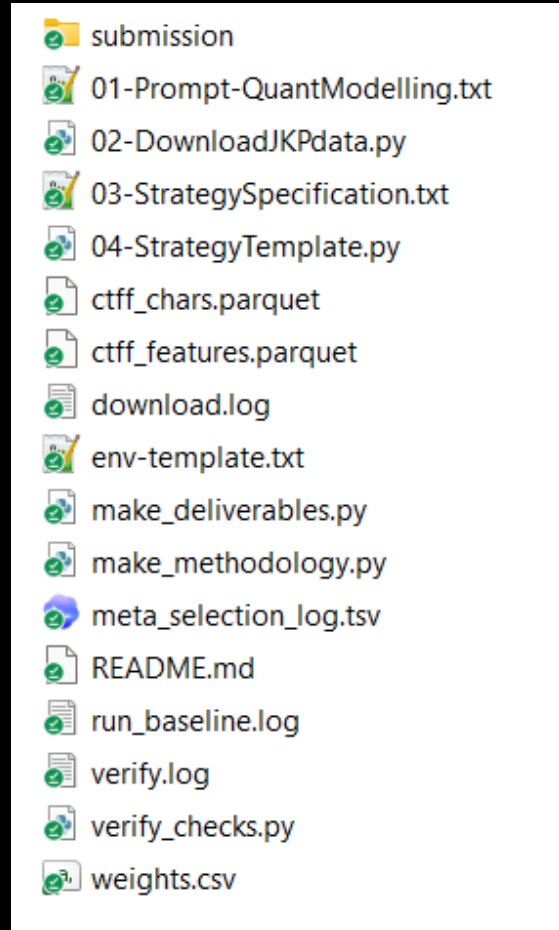
What the four stages did, plainly:

close to focus or unfocus Claude

+ [] verify_checks.py Auto mode

Ln 1, Col 1 Spaces: 4 UTF-8 LF {} Python Python 3.12.13.final.0

In Windows File Explorer (not VS Code) open the /submission/methodology.pdf



Compare to current leaderboard

Rank	Model Name	Author	Language	Files	Sharpe Ratio	Annual Return	Volatility	Max Drawdown	Year
1	Implementation of the Nonlinear Portfolio Transformer Submitted Apr 3, 2026 · Scored Apr 4, 2026	Gareth Campbell	Python	 	2.7879	27.88%	10.00%	-18.74%	2024
2	LightGBM + Investment Families Submitted Apr 22, 2026 · Scored Apr 28, 2026	Darren Li	Python		2.1854	21.85%	10.00%	-19.40%	2024
3	MAXSER Submitted Jan 13, 2026 · Scored Jan 13, 2026	Yingying Li	R	 	2.0261	20.26%	10.00%	-13.23%	2024
4	Benchmark: Instrumented PCA factor portfolio  Benchmark	—	Python		1.9481	19.48%	10.00%	-9.92%	2024
5	ClusterSharpe Selection + MeanVar Weighting + Linear LS Submitted Mar 10, 2026 · Scored Mar 11, 2026	Gareth Campbell	Python		1.5010	15.01%	10.00%	-20.71%	2024
6	Benchmark: Kozak-Nagel-Santosh rank-weighted factors portfolio  Benchmark	—	R		1.4195	14.20%	10.00%	-8.12%	2024
7	Bull/Bear Naive Submitted Feb 20, 2026 · Scored Feb 24, 2026	Kavita Kar	Python	 	1.3762	13.76%	10.00%	-23.86%	2024
8	Cross-Sectional Return Ensemble (CSRE) Submitted Feb 21, 2026 · Scored Feb 23, 2026	Anna Siamionava	Python	 	1.2591	12.59%	10.00%	-29.00%	2024
9	Rolling Factor Selection with RankSharpe Weighting Submitted Mar 9, 2026 · Scored Mar 10, 2026	Gareth Campbell	Python		1.1895	11.89%	10.00%	-15.19%	2024
10	Grid-Optimized Group Lasso (GOGL) Submitted Feb 19, 2026 · Scored Feb 20, 2026	mbn	Python		1.1030	11.03%	10.00%	-29.37%	2024

Challenge

- Ask Claude for adjustments to the strategy which could maximise the Sharpe Ratio further
- Upload to Canvas for the assignment
- Upload to <https://jkpfactors.com/ctf/submit> to the global leaderboard