

# Contents

Tutorial 1: VS Code Basics for Complete Beginners	1
□ What You'll Learn (60 minutes)	1
Part 1: The VS Code Interface (10 minutes)	1
Part 2: Creating Your First Project (10 minutes)	2
Part 3: Essential Keyboard Shortcuts (15 minutes)	3
Part 4: Working with Terminal (10 minutes)	4
Part 5: Writing and Running Python Code (15 minutes)	5
Part 6: File Management Tricks (5 minutes)	6
Part 7: Search and Replace (5 minutes)	6
Part 8: Customization (5 minutes)	6
Part 9: Command Palette (The Secret Weapon)	7
Part 10: Practice Project (5 minutes)	7
□ Skills Checklist	8
□ Keyboard Shortcuts Summary Card	9
□ What's Next?	9
□ Bonus: VS Code Tips	9

## Tutorial 1: VS Code Basics for Complete Beginners

### □ What You'll Learn (60 minutes)

By the end of this tutorial, you'll: - Navigate VS Code like a pro - Use keyboard shortcuts for speed - Manage files and folders - Use the integrated terminal - Write and run your first Python code - Feel confident in VS Code

Perfect for: Never used VS Code before Time: 60 minutes hands-on Prerequisites: VS Code installed (see QUICK\_START\_GUIDE.md)

---

### Part 1: The VS Code Interface (10 minutes)

#### Open VS Code

1. Open VS Code from Desktop or Start Menu
2. You should see the Welcome screen

#### The 5 Key Areas

Menu Bar (File, Edit, View, etc.)

	EDITOR AREA	S
S	(Your code goes here)	I
I		D
D		E
E		
B		B
A		A
R		R

PANEL (Terminal, Problems, Output)

1. Activity Bar (Left edge) - Explorer (files) ☐ - Search ☐ - Source Control (Git) ☐ - Run & Debug ☐ - Extensions ☐
2. Side Bar (Left panel) - Shows content based on Activity Bar selection - Can be hidden with `Ctrl+B`
3. Editor Area (Center) - Where you write code - Can have multiple files open in tabs - Split into multiple panes
4. Panel (Bottom) - Terminal ☐ - Problems (errors) ☐ - Output ☐ - Debug Console ☐
5. Status Bar (Bottom edge) - Line/column number - File type - Git branch - Errors/warnings

☐ Try It: Explore the Interface

1. Click each icon in Activity Bar
  2. See what appears in Side Bar
  3. Toggle Side Bar: Press `Ctrl+B`
  4. Toggle Panel: Press `Ctrl+J`
  5. Toggle both: Get clean workspace!
- 

## Part 2: Creating Your First Project (10 minutes)

Create Project Folder

Method 1: Using Explorer 1. Click Explorer icon (☐) in Activity Bar 2. Click "Open Folder" 3. Navigate to Documents 4. Click "New Folder" → Name it: `financial-modeling-practice` 5. Click "Select Folder"

Method 2: Using Terminal (Pro way!) 1. Press `Ctrl+``` (backtick) to open terminal 2. Type:  
``bash cd Documents mkdir financial-modeling-practice cd financial-modeling-practice  
`` 3. File → Open Folder → Select `financial-modeling-practice`

Create Your First File

3 Ways to Create a File:

Method 1: Side Bar - Hover over folder name in Explorer - Click "New File" icon (☐) - Name: `my_first_code.py`

Method 2: Keyboard (Fastest!) - Press `Ctrl+N` (New file) - Press `Ctrl+S` (Save) - Name: `calculator.py`

Method 3: Quick Open - Press `Ctrl+P` - Type: `calculator.py` - If doesn't exist, it'll create it!

☐ Try It: Create These Files

Create this structure:

```
financial-modeling-practice/  
  hello.py  
  calculator.py  
  notes.txt
```

Practice: 1. Create `hello.py` using Method 1 2. Create `calculator.py` using Method 2 3. Create `notes.txt` using Method 3

---

## Part 3: Essential Keyboard Shortcuts (15 minutes)

### The Most Important Shortcut

**Ctrl+P** - Quick Open - Opens any file instantly - Just type part of filename - This will change your life!

☐ Try it: 1. Press Ctrl+P 2. Type `hello` 3. Press Enter 4. File opens!

### File Operations

Ctrl+N	New file
Ctrl+O	Open file
Ctrl+S	Save file
Ctrl+Shift+S	Save as
Ctrl+W	Close current file
Ctrl+K W	Close all files
Ctrl+Tab	Switch between open files

☐ Practice Drill: 1. Ctrl+N - Create new file 2. Type some text 3. Ctrl+S - Save it 4. Ctrl+W - Close it 5. Ctrl+P - Reopen it!

### Editing Shortcuts

Ctrl+C	Copy line (no selection needed!)
Ctrl+X	Cut line
Ctrl+V	Paste
Ctrl+Z	Undo
Ctrl+Y	Redo
Ctrl+/	Comment/uncomment line

☐ Try this amazing trick: 1. Open `calculator.py` 2. Type: `print("Hello")` 3. Don't select anything! 4. Press Ctrl+C - Copies whole line! 5. Press Ctrl+V - Pastes it! 6. Press Ctrl+/ - Comments it out!

### Moving Code

Alt+↑	Move line up
Alt+↓	Move line down
Shift+Alt+↑	Copy line up
Shift+Alt+↓	Copy line down

☐ Practice: 1. Write 3 lines of code 2. Use Alt+↑/↓ to rearrange 3. Use Shift+Alt+↓ to duplicate

### Selection Shortcuts

Ctrl+D	Select next occurrence
Ctrl+L	Select entire line
Shift+Alt+→	Expand selection

☐ Multi-cursor magic: 1. Type: `value = 100` 2. Press Ctrl+D - Selects "value" 3. Press Ctrl+D again - Selects next "value" 4. Type to change all at once!

### Navigation

Ctrl+G	Go to line number
Ctrl+Home	Go to start of file
Ctrl+End	Go to end of file
Ctrl++/→	Jump by word

---

## Part 4: Working with Terminal (10 minutes)

### Open Terminal

3 Ways: 1. Press Ctrl+`` (backtick under Esc) 2. Menu: View → Terminal 3. Ctrl+Shift+P → Type "terminal" → Select "Toggle Terminal"

### Terminal Basics

The terminal lets you run commands:

*# See where you are*

`pwd`

*# List files*

`dir` (Windows CMD)

`ls` (PowerShell/Git Bash)

*# Change directory*

`cd` folder-name

*# Go up one level*

`cd ..`

*# Create folder*

`mkdir` new-folder

*# Create file*

`echo.` > newfile.txt (CMD)

New-Item newfile.txt (PowerShell)

### Multiple Terminals

- Click + to create new terminal
- Click dropdown to switch between them
- Use different terminals for different tasks:
  - Terminal 1: Running Python scripts
  - Terminal 2: Git commands
  - Terminal 3: Installing packages

### ☐ Practice: Terminal Commands

1. Open terminal ('Ctrl+')
  2. Check location: `pwd`
  3. List files: `dir` or `ls`
  4. Create folder: `mkdir` practice
  5. Navigate into it: `cd` practice
  6. Create file: `echo print("Test") > test.py`
  7. Go back: `cd ..`
-

## Part 5: Writing and Running Python Code (15 minutes)

### Your First Python Program

1. Create new file: hello\_world.py
2. Type this code:

```
# My first Python program
print("Hello, World!")
print("I'm learning VS Code!")

# Simple calculation
investment = 10000
rate = 0.08
years = 5

future_value = investment * (1 + rate) ** years

print(f"Investment: ${investment:,.2f}")
print(f"Rate: {rate:.1%}")
print(f"Future Value: ${future_value:,.2f}")
```

### Run Your Code

Method 1: Play Button - Look for ► (Run button) in top right - Click it!

Method 2: Keyboard (Pro way!) - Press Ctrl+F5

Method 3: Terminal

```
python hello_world.py
```

### □ Your Turn: Build a Calculator

Create simple\_calculator.py:

```
# Simple Financial Calculator

print("=== Financial Calculator ===\n")

# Present Value Calculator
print("Present Value Calculator")
future_value = float(input("Enter future value: $"))
rate = float(input("Enter discount rate (as decimal, e.g., 0.10): "))
years = int(input("Enter number of years: "))

present_value = future_value / (1 + rate) ** years

print(f"\nPresent Value: ${present_value:,.2f}")
print(f"Discount: ${future_value - present_value:,.2f}")
```

Run it (Ctrl+F5) and test with: - Future value: 10000 - Rate: 0.10 - Years: 5

---

## Part 6: File Management Tricks (5 minutes)

### Split Editor

Work on two files side-by-side:

Method 1: - Right-click file tab → “Split Right”

Method 2: - Ctrl+\ (splits current file)

Method 3: - Drag file tab to the right edge

□ Try it: 1. Open calculator.py and hello.py 2. Split them side-by-side 3. Edit both at once!

### Tab Groups

Ctrl+1	Focus first editor group
Ctrl+2	Focus second editor group
Ctrl+3	Focus third editor group
Ctrl+W	Close current tab
Ctrl+K Ctrl+W	Close all tabs

### File Explorer Tricks

Right-click → Reveal in File Explorer  
(Opens Windows Explorer)

Click and drag files to rearrange

Ctrl+Click      Select multiple files

---

## Part 7: Search and Replace (5 minutes)

### Find in File

Ctrl+F	Find in current file
Ctrl+H	Find and replace
F3	Find next
Shift+F3	Find previous

### Find in All Files

Ctrl+Shift+F	Search across all files
Ctrl+Shift+H	Replace across all files

□ Practice: 1. Open calculator.py 2. Press Ctrl+F 3. Search for: “value” 4. Press Ctrl+H 5. Replace “value” with “amount” 6. Click “Replace All”

---

## Part 8: Customization (5 minutes)

### Change Theme

1. Press Ctrl+K Ctrl+T
2. Select a theme (try “Dark+ (default dark)”)

3. Pick what you like!

Popular themes: - Dark+ (default) - Light+ (default) - Monokai - Solarized Dark

### Change Font Size

1. Press Ctrl+, (Settings)
2. Search: "font size"
3. Change to 14 or 16
4. Makes code easier to read!

### Zoom

Ctrl+=	Zoom in
Ctrl+-	Zoom out
Ctrl+0	Reset zoom

---

## Part 9: Command Palette (The Secret Weapon)

What is Command Palette?

Press Ctrl+Shift+P - Access EVERYTHING in VS Code!

Can't remember a shortcut? Use Command Palette!

☐ Try these commands:

1. Ctrl+Shift+P
2. Type: "reload" → Select "Developer: Reload Window"
3. Ctrl+Shift+P
4. Type: "format" → Select "Format Document"
5. Ctrl+Shift+P
6. Type: "settings" → Select "Preferences: Open Settings"

Most used commands: - "Format Document" - Clean up code - "Change Language Mode" - Change file type  
- "Transform to Uppercase" - Change case - "Sort Lines Ascending" - Sort lines

---

## Part 10: Practice Project (5 minutes)

Build a Simple Finance Tool

Create `investment_calculator.py`:

```
"""
Simple Investment Calculator
Practice for VS Code skills
"""

def calculate_future_value(principal, rate, years):
    """Calculate future value of investment"""
    return principal * (1 + rate) ** years

def calculate_compound_annual_growth_rate(beginning, ending, years):
    """Calculate CAGR"""
```

```

    return (ending / beginning) ** (1 / years) - 1

# Main program
print("Investment Analysis Tool")
print("=" * 40)

# Get user input
principal = float(input("Initial investment: $"))
annual_rate = float(input("Annual return (e.g., 0.08 for 8%): "))
years = int(input("Investment period (years): "))

# Calculate
future = calculate_future_value(principal, annual_rate, years)
total_return = future - principal
cagr = calculate_compound_annual_growth_rate(principal, future, years)

# Display results
print("\n" + "=" * 40)
print("RESULTS")
print("=" * 40)
print(f"Initial Investment:   ${principal:,.2f}")
print(f"Future Value:           ${future:,.2f}")
print(f"Total Return:           ${total_return:,.2f}")
print(f"CAGR:                   {cagr:.2%}")
print(f"Multiple:               {future/principal:,.2f}x")

```

Practice these VS Code skills:

1. File creation: Ctrl+N to create file
  2. Save: Ctrl+S to save
  3. Run: Ctrl+F5 to test
  4. Format: Ctrl+Shift+P → "Format Document"
  5. Comment: Select lines, Ctrl+/
  6. Duplicate: Shift+Alt+↓ to copy lines
- 

## ☐ Skills Checklist

After this tutorial, you should be able to:

- ☐ Navigate VS Code interface confidently
  - ☐ Create and organize files/folders
  - ☐ Use essential keyboard shortcuts
  - ☐ Open and use integrated terminal
  - ☐ Write and run Python code
  - ☐ Search and replace text
  - ☐ Split editor for multiple files
  - ☐ Use Command Palette
  - ☐ Customize VS Code to your liking
  - ☐ Feel comfortable in VS Code!
-



## ☐ Keyboard Shortcuts Summary Card

Print this or keep it visible:

### VS CODE ESSENTIALS (Windows)

#### FILES

Ctrl+N	New file
Ctrl+P	Quick open (MOST IMPORTANT!)
Ctrl+S	Save
Ctrl+W	Close file

#### EDITING

Ctrl+/	Comment/uncomment
Alt+↑/↓	Move line
Shift+Alt+↓	Copy line
Ctrl+D	Select next match

#### NAVIGATION

Ctrl+G	Go to line
Ctrl+P	Find file
Ctrl+Shift+F	Find in all files

#### VIEW

Ctrl+B	Toggle sidebar
Ctrl+J	Toggle panel
Ctrl+`	Toggle terminal

#### POWER MOVES

Ctrl+Shift+P	Command Palette (ACCESS EVERYTHING!)
Ctrl+\	Split editor
Ctrl+1/2/3	Switch editor groups

---

## ☐ What's Next?

You're now comfortable with VS Code!

Next steps: 1. ☐ Practice shortcuts for 2-3 days 2. ☐ Continue to: [Tutorials/02\\_GitHub\\_Copilot\\_Hands\\_On.md](#)  
3. ☐ Start using VS Code for all your coding

Pro tip: Use one new shortcut every day. In a week, you'll be 10x faster!

---

## ☐ Bonus: VS Code Tips

Tip 1: Breadcrumbs

- See your location in file
- Click to navigate quickly
- Toggle: View → Show Breadcrumbs

#### Tip 2: Minimap

- See overview of entire file
- Click to jump to sections
- Disable in Settings if you don't like it

#### Tip 3: Auto Save

- File → Auto Save
- Never lose work again!

#### Tip 4: Zen Mode

- View → Appearance → Zen Mode
- Full screen, distraction-free
- Exit: Press `Esc` twice

---

☐ Congratulations! You've mastered VS Code basics!

Time to learn GitHub and Copilot: → [Tutorials/02\\_GitHub\\_Copilot\\_Hands\\_On.md](#)

---

Estimated completion time: 60 minutes Difficulty: Beginner Next: GitHub & Copilot