

Install Odoo 20 on Windows

Source install + VS Code debugging, step by step · Odooistic companion guide · tested October 2026



What you will have at the end

- Odoo 20 running from source at `http://localhost:8069`
- A private Python 3.12 virtual environment just for Odoo 20
- VS Code: press F5 to run Odoo, set breakpoints and step through Odoo's own code
- A `custom_addons` folder for your own modules
- Works next to other Python and PostgreSQL versions you may already have

Versions used (and why)

Item	Version	Why this one
Python	3.12.10	Odoo 20 accepts 3.12–3.14, but on Windows only 3.12 installs every package from ready-made builds. 3.12.10 is the last 3.12 with a Windows installer.
PostgreSQL	16	Odoo 20 minimum is 16 (<code>MIN_PG_VERSION = 16</code> in <code>release.py</code>). Matches Ubuntu 24.04 servers.
Git	Latest	To download Odoo's source from GitHub.
wkhtmltopdf	0.12.6	The version Odoo recommends for PDF reports. Look for 'with patched qt'.
VS Code	Latest	With the Python extension by Microsoft (includes Pylance and the Python Debugger).
C++ Build Tools	Not needed	With Python 3.12 every Odoo 20 package installs without a compiler.

Tip: Use PowerShell for every command in this guide. After any install that changes your PATH, open a **new** PowerShell window – windows that were already open never see the change.

The plan

• ODOOISTIC

THE PLAN

Install Odoo 20 on Windows

From source, with VS Code debugging · step by step

1 Prerequisites

Python 3.12.10
PostgreSQL 16
Git
wkhtmltopdf 0.12.6
VS Code + Python ext

2 Odoo 20 Source

C:\Odoo20 folder
git clone 20.0
venv + requirements
odoo20 DB user
odoo.conf

3 VS Code

Select interpreter
launch.json + settings
F5 first run
Create database
Breakpoint debugging

Part 1: Prerequisites

Step 1. Python 3.12.10

• ODOOISTIC

PART 1 · PREREQUISITES

1 Install Python 3.12

python.org → Downloads → Windows →
3.12.10

- Use the **Windows installer (64-bit)** from the table
- Skip the yellow "install manager" button (it gives 3.14)
- Tick **Add python.exe to PATH**
- Already have 3.13? Fine – use `py -3.12`

DON'T MISS THIS BOX

Python 3.12.10 (64-bit) Setup

Install Python 3.12.10 (64-bit)

Select Install Now to install Python with default settings.

☒ Use admin privileges when installing py.exe

☒ Add python.exe to PATH

Customize installation

Install Now

Windows PowerShell

```
PS C:\Odoo20> py --list
-V:3.13 *      Python 3.13 (64-bit)
-V:3.12       Python 3.12 (64-bit)
PS C:\Odoo20> py -3.12 --version
Python 3.12.10
```

- Go to `python.org/downloads`. **Don't** click the yellow 'Download Python install manager' button (it installs 3.14).
- Click **Windows** under 'Looking for Python with a different OS?', press Ctrl+F and search 3.12.10.

3. On the 3.12.10 page, ignore the 'superseded by 3.12.15' note (later 3.12 releases have no Windows installer). Scroll to **Files** and click **Windows installer (64-bit)** in the table (not the install-manager button in the Windows box).
4. Run `python-3.12.10-amd64.exe`. Tick **Add python.exe to PATH**, then **Install Now**.
5. Click **Disable path length limit** if offered, then Close.
6. Open a new PowerShell and check:

```
py --list
py -3.12 --version      # Python 3.12.10
```

Tip: If `python --version` still shows another version (for example 3.13.7), that is normal when you already had Python installed. Don't uninstall anything – use `py -3.12`. Inside the virtual environment, `python` will be 3.12 automatically.

• ODOOISTIC

PART 1 • WHY THIS VERSION?

The newest Python isn't the **right** Python

Odoo 20 accepts Python 3.12–3.14, but on **Windows** some pinned packages have no ready-made build for newer versions.

Python	Result on Windows	Why
3.14	✗ Usually fails	<code>psycopg2==2.9.10</code> has no Windows wheel for 3.14 → pip tries to compile it
3.13	⚠ Needs a compiler	<code>r1-renderPM</code> and <code>MarkupSafe</code> pins have no Windows wheels for 3.13
3.12.10	✓ Installs cleanly	Every package installs from a ready-made wheel – no C++ Build Tools needed

Checked against `odoo/odoo 20.0 requirements.txt` and PyPI wheel lists

Step 2. PostgreSQL 16

The screenshot shows the Odoo 20 installation prerequisites page on the left and the PostgreSQL 16 setup wizard on the right. The prerequisites page includes a yellow circle with the number 2, the title 'Install PostgreSQL 16', and a list of instructions: 'Odoo 20 minimum is 16 (MIN_PG_VERSION = 16)', 'Untick Stack Builder', 'Older PostgreSQL on 5432? New one gets 5433', and 'Write down the postgres password & port'. The setup wizard has two screens. The first screen, 'Select Components', shows checkboxes for 'PostgreSQL Server', 'pgAdmin 4', 'Stack Builder' (which is highlighted with a yellow bar and a red 'UNTICK' button), and 'Command Line Tools'. The second screen, 'Port', shows a text box with '5433' and a yellow callout box that says 'REMEMBER THIS PORT'.

Check what you already have first:

```
Get-Service *postgres*
Get-ChildItem "C:\Program Files\PostgreSQL"
```

Version 15 or lower is too old for Odoo 20. Leave an old version in place if other projects use it – 16 installs alongside it.

1. Go to [postgresql.org/download/windows](https://www.postgresql.org/download/windows) and click **Download the installer** (it is hosted by EDB). Ignore any beta banner.
2. In the EDB table, find **16** and click the download icon under **Windows x86-64**.
3. Run it. Components: keep PostgreSQL Server, pgAdmin 4, Command Line Tools; **untick Stack Builder**.
4. Set a password for the postgres superuser and write it down.
5. Port: keep what it offers. It is **5432** on a clean PC, or **5433** if another PostgreSQL already uses 5432. Write it down.
6. Default locale, Next, Install, Finish (untick 'Launch Stack Builder').

Add PostgreSQL to your PATH (see the PATH fix below), then in a **new** PowerShell:

```
psql --version
psql -U postgres -p 5433 -c "select version();" # use your port
```

Confirm the port at any time:

```
Select-String -Path "C:\Program Files\PostgreSQL\16\data\postgresql.conf" -Pattern "^port"
```

The PATH fix ('... is not recognized')

ODOOISTIC

PART 1 · THE PATH FIX

Fix the PATH

"Not recognised" after installing? Windows just doesn't know where to look.

- Start → **Edit environment variables for your account**
- Path → Edit → **New** → paste the bin folder
- Then open a **NEW** terminal – old windows never see it
- No admin rights needed for the user PATH

Edit environment variable

Path (user)

- C:\Users\you\AppData\Local\Programs\Python\Python312\
- C:\Users\you\AppData\Local\Programs\Python\Python312\Scripts\
- C:\Program Files\PostgreSQL\16\bin
- C:\Program Files\wkhtmltopdf\bin

New Edit OK

Windows PowerShell

```
PS C:\0doo20> Test-Path "C:\Program Files\wkhtmltopdf\bin\wkhtmltopdf"
True
# ...add to PATH, then open a NEW PowerShell window
PS C:\0doo20> wkhtmltopdf --version
wkhtmltopdf 0.12.6 (with patched qt)
```

The PostgreSQL and wkhtmltopdf installers don't add themselves to PATH. Add their bin folders to your **user** PATH (no admin rights needed):

GUI: Start → type 'environment' → **Edit environment variables for your account** → Path → Edit → New → paste the folder → OK, OK.

PowerShell:

```
$p = [Environment]::GetEnvironmentVariable("Path", "User")
[Environment]::SetEnvironmentVariable("Path", "$p;C:\Program Files\PostgreSQL\16\bin", "User")
```

Watch out: 'Requested registry access is not allowed' means you tried the **Machine** (system) PATH without admin rights. Use "User" as above, or run PowerShell as Administrator.

Then close the window and open a **new** PowerShell. To refresh the current window instead (keeps an active venv):

```
$env:Path = "C:\Program Files\PostgreSQL\16\bin;" + $env:Path
```

Step 3. Git

• ODOOISTIC

PART 1 · PREREQUISITES

3

Install Git

git-scm.com → Downloads → Windows → x64 Setup

- Default editor: **Notepad** (not Vim)
- Initial branch: override to **main**
- PATH: **Git from the command line and 3rd-party software**
- Turn on long paths for Odoo's deep folders

Git Setup

Adjusting your PATH environment

How would you like to use Git from the command line?

- ☐ Use Git from Git Bash only
- ☒ **Git from the command line and also from 3rd-party software**
- ☐ Use Git and optional Unix tools from the Command Prompt

Back Next

Windows PowerShell

```
PS C:\Odoo20> git --version
git version 2.x.windows.1
PS C:\Odoo20> git config --global core.longpaths true
PS C:\Odoo20> git config --global user.name "Your Name"
PS C:\Odoo20> git config --global core.editor "code --wait"
```

1. Go to `git-scm.com` → Downloads → Windows → **Git for Windows/x64 Setup**.
2. Keep defaults except: default editor **Notepad**; initial branch **override** → **main**; PATH **Git from the command line and also from 3rd-party software**.
3. Then, in a new PowerShell:

```
git --version
git config --global core.longpaths true
git config --global user.name "Your Name"
git config --global user.email "you@example.com"
```

Step 4. wkhtmltopdf 0.12.6

• ODOOISTIC

PART 1 • PREREQUISITES

4

Install wkhtmltopdf

The tool Odoo uses for every PDF report - version **0.12.6**

- Download the Windows **64-bit** installer (0.12.6-1)
- Default folder: **C:\Program Files\wkhtmltopdf**
- Installer doesn't add itself to PATH – add the **bin** folder
- Look for “**with patched qt**”

Windows PowerShell

```
PS C:\0doo20> wkhtmltopdf --version
wkhtmltopdf : The term 'wkhtmltopdf' is not recognized...

# Installed? Check the file is there:
PS C:\0doo20> Test-Path "C:\Program Files\wkhtmltopdf\bin\wkhtmltopdf.exe"
True

# Add bin folder to your user PATH (no admin needed)
PS C:\0doo20> $p = [Environment]::GetEnvironmentVariable("Path","User")
PS C:\0doo20> [Environment]::SetEnvironmentVariable("Path",
PS C:\0doo20> "$p;C:\Program Files\wkhtmltopdf\bin", "User")

# Open a NEW PowerShell window
PS C:\0doo20> wkhtmltopdf --version
wkhtmltopdf 0.12.6 (with patched qt)
```

NEW TERMINAL!

1. Download the Windows 64-bit installer for **0.12.6** (wkhtmltox-0.12.6-1.msvc2015-win64.exe) from wkhtmltopdf.org/downloads.html (links may point to GitHub).
2. Install with defaults to C:\Program Files\wkhtmltopdf.
3. Check the file exists, add its bin folder to PATH, open a new PowerShell:

```
Test-Path "C:\Program Files\wkhtmltopdf\bin\wkhtmltopdf.exe"      # True
$p = [Environment]::GetEnvironmentVariable("Path","User")
[Environment]::SetEnvironmentVariable("Path", "$p;C:\Program Files\wkhtmltopdf\bin", "User")
# NEW PowerShell window:
wkhtmltopdf --version      # wkhtmltopdf 0.12.6 (with patched qt)
```

Step 5. VS Code

The screenshot shows a dark-themed interface for installing VS Code. On the left, a sidebar with a purple background contains the text 'ODOOISTIC', 'PART 1 • PREREQUISITES', and a large yellow circle with the number '5'. Below this, the title 'Install VS Code' is displayed in white and yellow, followed by the URL 'code.visualstudio.com → User Installer x64'. A list of instructions is provided: 'Tick both Open with Code options', 'Keep Add to PATH ticked', 'Extensions: Python by Microsoft (+ Pylance, Debugger)', and 'Optional: XML by Red Hat for Odoo views'. On the right, a 'Setup - Visual Studio Code' window is shown with the title 'Select Additional Tasks'. It asks 'Which additional tasks should be performed?' and lists four options, all of which are checked: 'Create a desktop icon', 'Add "Open with Code" action to file context menu', 'Add "Open with Code" action to directory context menu', 'Register Code as an editor for supported file types', and 'Add to PATH (requires shell restart)'. 'Back' and 'Next' buttons are at the bottom right of the window. Below the setup window, a Windows PowerShell terminal window is shown with the following commands and output: 'code --version' returns '1.x.x', and 'code --install-extension ms-python.python' returns 'Extension 'ms-python.python' was successfully installed.'

1. Go to `code.visualstudio.com` → **Download for Windows** (User Installer x64, no admin needed).
2. On **Select Additional Tasks**, tick both 'Open with Code' options, 'Register Code as an editor' and **Add to PATH**.
3. In VS Code press `Ctrl+Shift+X` and install **Python** by Microsoft (brings Pylance and Python Debugger).
Optional: **XML** by Red Hat.
4. In a new PowerShell:

```
code --version
git config --global core.editor "code --wait"
```

No C++ Build Tools needed

• ODOOISTIC

PART 1 • GOOD NEWS

Skip the ~~5 GB C++ Build Tools~~

Because we picked **Python 3.12**, every Odoo 20 package installs from a ready-made Windows build.

✓ Pre-built wheels

psycpg2 · lxml · Pillow
cryptography · reportlab
libsass · psutil · MarkupSafe

✓ Pure-Python fallbacks

ofxparse · vobject
rjsmin (falls back automatically)

Only if you see...

"Microsoft Visual C++ 14.0 or greater
is required"
...check your venv is really Python 3.12.

With Python 3.12, every package in Odoo 20's `requirements.txt` installs from a ready-made Windows build, or falls back to pure Python. Only install the Visual Studio Build Tools if you ever see 'Microsoft Visual C++ 14.0 or greater is required' – and if you do, first check your venv really is Python 3.12.

Part 2: Install Odoo 20 from source

Step 6. Project folder

• ODOOISTIC

PART 2 • ODOO 20 SOURCE

6

Project folder

One short folder at the root of C: – short paths avoid Windows errors.

- `odoo\` – Odoo's source (never edit)
- `custom_addons\` – your modules
- `.venv\` – private Python 3.12
- `.vscode\` sits next to them, **not** inside `odoo\`

File Explorer — C:\Odoo20

```
C:\Odoo20\
├── .vscode\
│   ├── launch.json      # F5 configs
│   └── settings.json    # IntelliSense
├── .venv\               # Python 3.12.10
├── custom_addons\       # your modules
│   └── hello_odoo20\
├── data\               # attachments
├── odoo\               # Odoo 20 source
│   ├── odoo-bin
│   └── requirements.txt
└── odoo.conf
```

```
mkdir C:\Odoo20
mkdir C:\Odoo20\custom_addons
cd C:\Odoo20
```

Tip: Keep the path short (root of C:). The `.vscode` folder goes directly inside `C:\Odoo20`, next to `odoo` and `.venv` – not inside `odoo\`. Windows ignores upper/lower case, so `C:\Odoo20` and `C:\odoo20` are the same folder.

Step 7. Clone Odoo 20

• ODOOISTIC

PART 2 · ODOO 20 SOURCE

7

Clone Odoo 20

Download the official 20.0 branch straight from GitHub.

- `--branch 20.0` – version 20
- `--depth 1` – latest snapshot only (hundreds of MB, not GBs)
- `odoo` – the target folder name

```
PS C:\Odoo20> mkdir C:\Odoo20
PS C:\Odoo20> mkdir C:\Odoo20\custom_addons
PS C:\Odoo20> cd C:\Odoo20
PS C:\Odoo20> git clone https://github.com/odoo/odoo.git --depth 1 --branch 20.0 odoo
Cloning into 'odoo'...
remote: Enumerating objects: 52xxx, done.
Receiving objects: 100% ... done.
Updating files: 100% ... done.
PS C:\Odoo20> dir odoo
    addons  debian  doc  odoo  setup
    odoo-bin  requirements.txt  setup.py ...
```

```
git clone https://github.com/odoo/odoo.git --depth 1 --branch 20.0 odoo
```

`--depth 1` downloads only the latest snapshot (hundreds of MB instead of several GB).

Step 8. Virtual environment and requirements

• ODOOISTIC

PART 2 · ODOO 20 SOURCE

8

venv + requirements

A private Python 3.12 just for this Odoo.

- Create with `py -3.12 -m venv .venv`
- Blocked script? `Set-ExecutionPolicy -Scope CurrentUser RemoteSigned`
- pip cache "Permission denied"? add `--no-cache-dir`
- Check: **Odoo Server 20.0**

```
PS C:\Odoo20> py -3.12 -m venv .venv
PS C:\Odoo20> .venv\Scripts\Activate.ps1
(.venv) PS C:\Odoo20> python --version
Python 3.12.10
(.venv) PS C:\Odoo20> python -m pip install --upgrade pip setuptools wheel
(.venv) PS C:\Odoo20> pip install --no-cache-dir -r odoo\requirements.txt
Successfully installed Babel-2.10.3 Jinja2-3.1.2 lxml-5.2.1 ...
psycopg2-2.9.9 reportlab-4.1.0 Werkzeug-3.0.1 ...
(.venv) PS C:\Odoo20> pip install watchdog
(.venv) PS C:\Odoo20> python odoo\odoo-bin --version
Odoo Server 20.0
```

```
cd C:\Odoo20
py -3.12 -m venv .venv
.venv\Scripts\Activate.ps1
python --version # Python 3.12.10
python -m pip install --upgrade pip setuptools wheel
pip install -r odoo\requirements.txt
pip install watchdog
python odoo\odoo-bin --version # Odoo Server 20.0
```

Watch out: 'running scripts is disabled on this system' → run `Set-ExecutionPolicy -Scope CurrentUser RemoteSigned` once, then activate again.

Watch out: '[Errno 13] Permission denied ... pip\cache' → run `pip install --no-cache-dir -r odoo\requirements.txt`. To clear the broken cache: close all terminals and run `Remove-Item "$env:LOCALAPPDATA\pip\cache" -Recurse -Force` (as Administrator if needed).

Step 9. Create the Odoo database user

• ODOOISTIC

PART 2 · ODOO 20 SOURCE

9

Create the DB user

Odoo refuses to run as the `postgres` superuser.

- Connect with the right port: `-p 5433`
- LOGIN = can connect
- CREATEDB = Odoo can create databases
- Simple password = local practice only

```
PS C:\Odoo20> psql -U postgres -p 5433 -c "select version();"
PostgreSQL 16.x, compiled by Visual C++ ...
PS C:\Odoo20> psql -U postgres -p 5433
postgres=# CREATE ROLE odoo20 WITH LOGIN CREATEDB PASSWORD 'odoo20';
CREATE ROLE
postgres=# \du
Role name | Attributes
-----+-----
odoo20    | Create DB
postgres  | Superuser, Create role, Create DB, ...
postgres=# \q
```

Odoo refuses to run as the postgres superuser. Use your own port in place of 5433:

```
psql -U postgres -p 5433 -c "select version();"      # must say PostgreSQL 16.x
psql -U postgres -p 5433
CREATE ROLE odoo20 WITH LOGIN CREATEDB PASSWORD 'odoo20';
\du
\q
```

pgAdmin alternative: Login/Group Roles → Create → name and password → Privileges: **Can login + Create databases**. A 'Console code page differs' warning from psql is harmless.

Step 10. odoo.conf

• ODOOISTIC

PART 2 · ODOO 20 SOURCE

10

Write **odoo.conf**

All settings in one file – no more long commands.

- **addons_path** – Odoo's modules + yours
- **data_dir** – keep files inside the project
- **db_port** – 5433 (or 5432 – what your installer showed)
- **admin_passwd** – protects the database manager

```
1 [options]
2 addons_path = C:\odoo20\odoo\addons,C:\odoo20\custom_addons
3 data_dir = C:\odoo20\data
4 db_host = localhost
5 db_port = 5433
6 db_user = odoo20
7 db_password = odoo20
8 admin_passwd = change-me-master-password
9 http_port = 8069
```

YOUR PORT

```
[options]
addons_path = C:\odoo20\odoo\addons,C:\odoo20\custom_addons
data_dir = C:\odoo20\data
db_host = localhost
db_port = 5433
db_user = odoo20
db_password = odoo20
admin_passwd = change-me-master-password
http_port = 8069
```

Tip: Running Odoo 19 on 8069 already? Set `http_port = 8070` and both can run together. The simple passwords here are for a local practice setup only.

Part 3: Run and debug in VS Code

Step 11. Open the project and wire up VS Code

ODOOISTIC

PART 3 · VS CODE

11

Wire up VS Code

Open `C:\Odoo20` → select the `.venv` interpreter
→ add two files.

- Debug – breakpoints (no auto-reload)**
- Run (auto-reload) – restarts on save**
- Update module – prompts for DB + module**
- justMyCode: false – step into Odoo core**

```
1 {
2   "version": "0.2.0",
3   "configurations": [
4     {
5       "name": "Odoo 20: Debug",
6       "type": "debugpy",
7       "request": "Launch",
8       "program": "${workspaceFolder}/odoo/odoo-bin",
9       "args": ["-c", "${workspaceFolder}/odoo.conf", "--dev=xml,qweb"],
10      "console": "integratedTerminal",
11      "justMyCode": false
12    },
13    // + "Run (auto-reload)" and "Update module"
14  ]
15 }
```

Status bar: Python 3.12.18 ('.venv': venv) .vscode/settings.json – python.a

1. **Close and reopen VS Code** so it picks up the new PATH entries. Then `cd C:\Odoo20; code .` – the Explorer header must say **ODOO20**.
2. `Ctrl+Shift+P` → **Python: Select Interpreter** → choose `.venv` (Python 3.12.10).
3. Create `.vscode\launch.json`:

```
{
  "version": "0.2.0",
  "configurations": [
    {
      "name": "Odoo 20: Debug",
      "type": "debugpy",
      "request": "launch",
      "program": "${workspaceFolder}/odoo/odoo-bin",
      "args": ["-c", "${workspaceFolder}/odoo.conf", "--dev=xml,qweb"],
      "console": "integratedTerminal",
      "justMyCode": false
    },
    {
      "name": "Odoo 20: Run (auto-reload)",
      "type": "debugpy",
      "request": "launch",
      "program": "${workspaceFolder}/odoo/odoo-bin",
      "args": ["-c", "${workspaceFolder}/odoo.conf", "--dev=all"],
      "console": "integratedTerminal",
      "justMyCode": false
    },
    {
      "name": "Odoo 20: Update module",
      "type": "debugpy",
      "request": "launch",
      "program": "${workspaceFolder}/odoo/odoo-bin",
      "args": ["-c", "${workspaceFolder}/odoo.conf", "-d", "${input:dbName}",
        "-u", "${input:moduleName}", "--dev=xml"],
      "console": "integratedTerminal",
    }
  ]
}
```

```
    "justMyCode": false
  }
],
"inputs": [
  { "id": "dbName", "type": "promptString", "description": "Database name", "default": "odoo20_demo" },
  { "id": "moduleName", "type": "promptString", "description": "Module to update", "default": "hello_" }
]
```

4. Create .vscode\settings.json:

```
{
  "python.defaultInterpreterPath": "${workspaceFolder}/.venv/Scripts/python.exe",
  "python.terminal.activateEnvironment": true,
  "python.analysis.extraPaths": ["${workspaceFolder}/odoo"],
  "files.exclude": { "**/__pycache__": true, "**/*.pyc": true }
}
```

Step 12. First run with F5

ODOOISTIC

PART 3 · VS CODE

12

Press F5

Run and Debug → Odoo 20: Debug → F5.
Read the log:

- Odoo version **20.0**
- database **odoo20@localhost:5433**
- Will use the **Wkhtmltopdf** binary = PDFs work
- HTTP running on **8069** → open the browser

```
INFO ? odoo: Odoo version 20.0
INFO ? odoo: Using configuration file at C:\odoo20\odoo.conf
INFO ? odoo: addons paths: [..., 'C:\\odoo20\\custom_addons']
INFO ? odoo: database: odoo20@localhost:5433
INFO ? odoo.addons.base.models.ir_actions_report: Will use the
Wkhtmltopdf binary at C:\Program Files\wkhtmltopdf\bin\wkhtmltopdf.exe
INFO ? odoo.service.server: HTTP service (werkzeug) running on ...:8069
```

localhost:8069/web/database/manager

Master Password	Database Name
.....	odoo20_demo
Email	Password
admin	
Language	Country
English (UK)	United Kingdom
☒ Demo data	Create database

1. Ctrl+Shift+D → choose **Odoo 20: Debug** → press **F5**. Allow Python on **Private networks** if Windows Firewall asks.
2. Check the log for: **Odoo version 20.0**, **database: odoo20@localhost:5433**, **Will use the Wkhtmltopdf binary at ...**, **HTTP service running on ...:8069**.
3. Open <http://localhost:8069> and fill in the database manager: master password = admin_passwd, database odoo20_demo, email admin, a password, English (UK), United Kingdom, tick Demo data → **Create database**.
4. PDF test: Sales → open a quotation → Print.

Step 13. Scaffold a module and hit a breakpoint

• ODOOISTIC

PART 3 · VS CODE

13

Hit a breakpoint

Scaffold a module, set a red dot, trigger it from the browser.

- odoo-bin scaffold hello_odoo20 custom_addons
- Uncomment models, views + access CSV
- Apps → Update Apps List → Activate
- F10 step over · F11 step into · F5 continue

```
3 from odoo import models, fields, api
4
5 class hello_odoo20(models.Model):
6     _name = "hello_odoo20.hello_odoo20"
7     name = fields.Char()
8     value = fields.Integer()
9     value2 = fields.Float(compute="_value_pc", store=True)
10
11     @api.depends("value")
12     def _value_pc(self):
13         for record in self:
14             record.value2 = float(record.value) / 100
```

PAUSED HERE

VARIABLES

Locals

```
self = hello_odoo20.hello_odoo20(<NewId origin=1>,)
record = hello_odoo20.hello_odoo20(<NewId>,)
value = 50
name = 'My first record'
```

```
python odoo\odoo-bin scaffold hello_odoo20 custom_addons
```

1. Uncomment the model in `models\models.py` (select, Ctrl+), the whole of `views\views.xml`, and the `'security/ir.model.access.csv'` line in `__manifest__.py`.
2. Click in the gutter on the `for record in self:` line inside `_value_pc` – a red dot appears.
3. Stop (Shift+F5) and start again (F5) with **Odoo 20: Debug**.
4. In Odoo: Apps → **Update Apps List** (turn on developer mode with `?debug=1` if it's missing) → remove the 'Apps' filter → search `hello_odoo20` → **Activate**.
5. Open the new menu, click New, type a value (e.g. 50) and tab out. VS Code pauses on your line. Inspect `self` in Variables. **F10** step over, **F11** step into, **F5** continue.

Troubleshooting

• ODOOISTIC

FIXES FOR THE ERRORS WE HIT

Top Windows errors & fixes

You see...	Fix
<code>python --version</code> still says 3.13	Normal – use <code>py -3.12</code> ; inside the venv it's 3.12
<code>psql / wkhtmltopdf</code> "not recognized"	Add its <code>bin</code> folder to your user PATH, then open a new terminal
"Requested registry access is not allowed"	Use the User PATH (no admin) or run PowerShell as Administrator
"running scripts is disabled on this system"	<code>Set-ExecutionPolicy -Scope CurrentUser RemoteSigned</code>
pip cache "Permission denied"	<code>pip install --no-cache-dir -r odoo\requirements.txt</code>
Connection refused / password failed	Check <code>db_port</code> (5433?) and the odoo20 password in <code>odoo.conf</code>
Breakpoint is a hollow grey circle	Use the Odoo 20: Debug config (auto-reload detaches the debugger)

Error / symptom	Fix
<code>python --version</code> shows 3.13 (or another version)	Normal if you already had Python. Use <code>py -3.12</code> ; inside the venv python is 3.12.
'psql' / 'wkhtmltopdf' is not recognized	Add its bin folder to your user PATH, then open a new PowerShell (or refresh <code>\$env:Path</code>).
'Requested registry access is not allowed'	Use the User PATH, or run PowerShell as Administrator for the system PATH.
'running scripts is disabled on this system'	<code>Set-ExecutionPolicy -Scope CurrentUser RemoteSigned</code>
pip '[Errno 13] Permission denied' in pip\cache	<code>pip install --no-cache-dir -r odoo\requirements.txt</code> , then clear the cache folder.
psycpg2 or 'Microsoft Visual C++ 14.0' errors	Your venv isn't Python 3.12. Delete <code>.venv</code> and recreate with <code>py -3.12 -m venv .venv</code> .
'connection refused' / 'could not connect'	Wrong port (check <code>db_port</code>) or the postgresql-x64-16 service is stopped (services.msc).
'password authentication failed for user odoo20'	<code>db_password</code> in <code>odoo.conf</code> must match the CREATE ROLE password.
'Using the database user postgres is a security risk'	Set <code>db_user = odoo20</code> .
'Address already in use' on 8069	Another Odoo is running. Stop it or set <code>http_port = 8070</code> .
No 'Will use the Wkhtmltopdf binary' line	VS Code was open before the PATH change – close it fully and reopen.
Breakpoint is a hollow grey circle	Use the Odoo 20: Debug config (auto-reload detaches the debugger).
Module missing from Apps	Update Apps List, clear the 'Apps' filter, check <code>custom_addons</code> is in <code>addons_path</code> .

Daily cheat sheet

```
cd C:\Odoo20
.venv\Scripts\Activate.ps1
python odoo\odoo-bin -c odoo.conf # run
python odoo\odoo-bin -c odoo.conf -d odoo20_demo -u hello_odoo20 # upgrade module
# or in VS Code: Ctrl+Shift+D -> Odoo 20: Debug -> F5
```

Watch the full walkthrough on the **Odooistic** YouTube channel, in the *Odoo Ops: Install, Deploy, Migrate* playlist.