**Types of Actions**

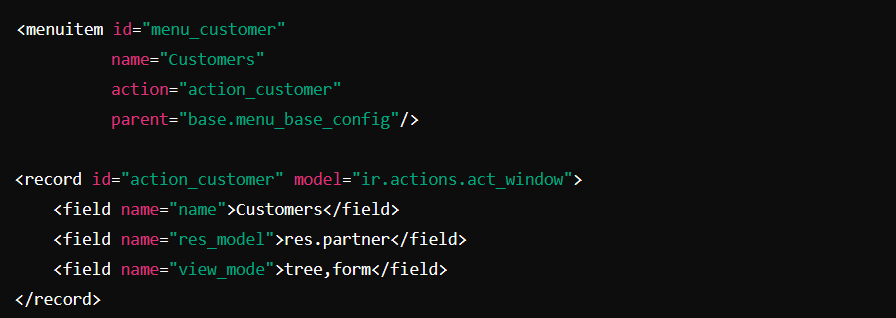
<https://youtu.be/ADG6DwpLDrw?si=ZiEsa3lPZxrlAPch>

**1. Window Action:**

A window action is used to navigate to a specific view of a model. It can open a list view, form view, kanban view, or any other defined view. Window actions are commonly triggered by **menus or buttons** and help users interact with records.

For example, clicking a **menu item** like "Customers" opens the list view of customer records.

**1️: Calling from a Menu Item (Menu Item + Action)** ✅



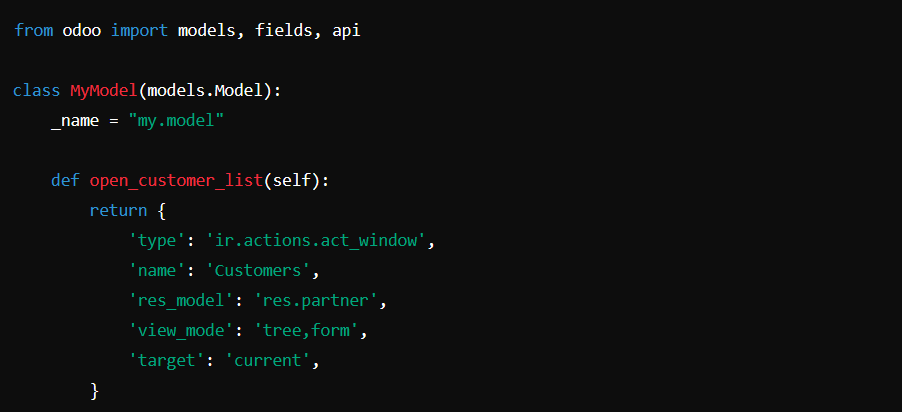
* The <menuitem> is defined with action="action\_customer", which links it to an action.
* The <record> of type ir.actions.act\_window defines an action that opens the res.partner model in tree and form views.
* **name**: The name of the action (visible in the UI).
* **res\_model**: Specifies the model to display (in this case, sale.order for Sales Orders).
* **view\_mode**: Defines the views to use (in this case, list (tree) and form views).
* **target**: The action target (current means it opens within the current window).
* **domain**: This filters records by state (only draft sales orders in this case).

**2️: Calling from a Button (Button + Python)** ✅

XML:

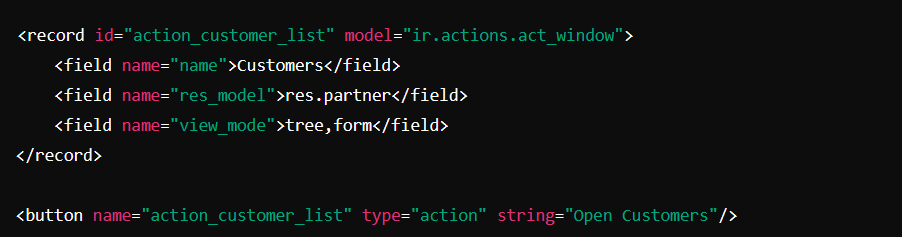


Python:



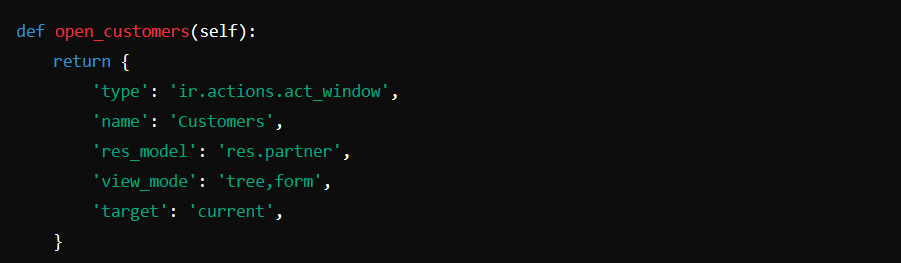
* The XML button calls the open\_customer\_list method.
* The Python method returns an ir.actions.act\_window action to open the res.partner model in tree and form views.

**3: Calling from a Button (Button + Action)** ✅



<button name="%(action\_custom\_list)d" string="Open Example" type="action"/>

**4: Calling from a Python Function ✅**

****

* This function can be called from a button or another function to open a specific model.

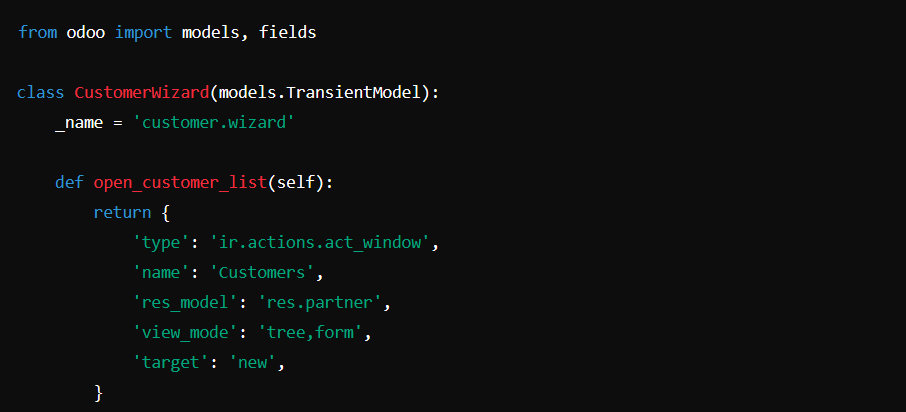
**5: Calling from JavaScript (do\_action)** ✅



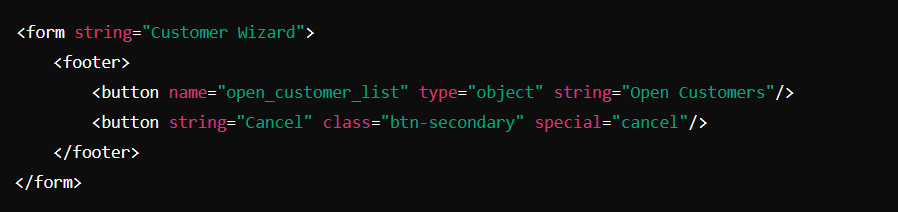
* do\_action(action) triggers the window action from JavaScript.
* This can be used in custom buttons or event handlers.

**6: Calling from a Wizard (Transient Model)** ✅

Wizard Model (Python)

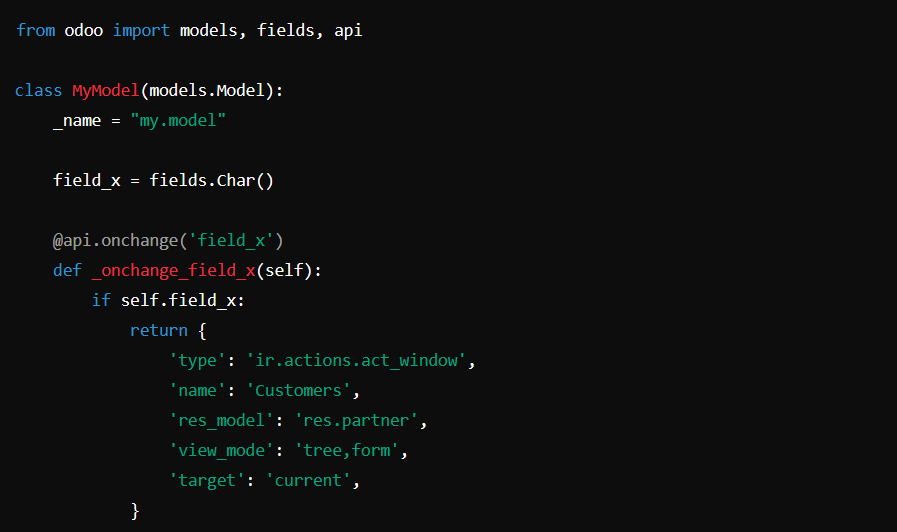


Wizard Form (XML)



* The wizard provides a button that triggers ir.actions.act\_window to open the customer list.

**7: Calling from onchange Method** ✅



* When field\_x changes, the onchange method triggers an action to open the res.partner view.

**8: Calling from an Automated** ✅

-------

**2. URL Action:**

A **URL action** in Odoo allows you to redirect users to an external or internal URL.



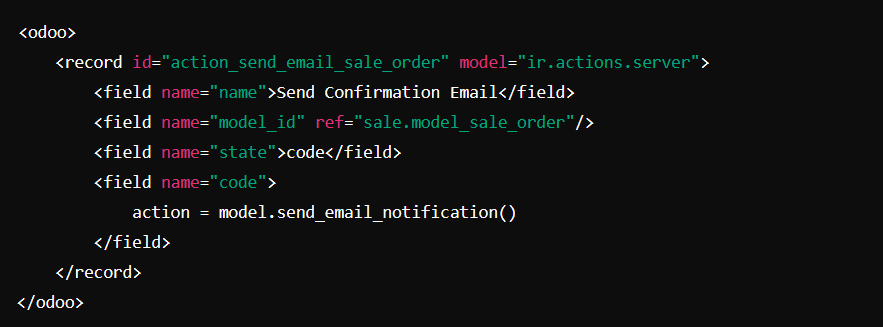
* **name**: Action name.
* **url**: The URL to open.
* **target**:
* **self** → Opens in the same tab.
* **new** → Opens in a new tab.

**3. Server Action**

Server Actions allow for executing Python code, sending emails, or calling methods on models.

**Example 1:** **Server Action to execute Python code**

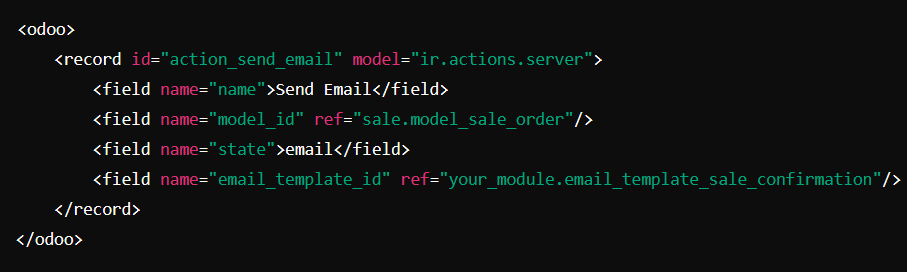
This server action sends an email when a Sales Order is confirmed.



* **state**: Set to code to execute custom Python code.
* **code**: The custom Python code to run (model.send\_email\_notification()).
* **model\_id**: The model in which the server action is executed.

**Example 2: Server Action to send email**

This server action uses an email template to send a message.

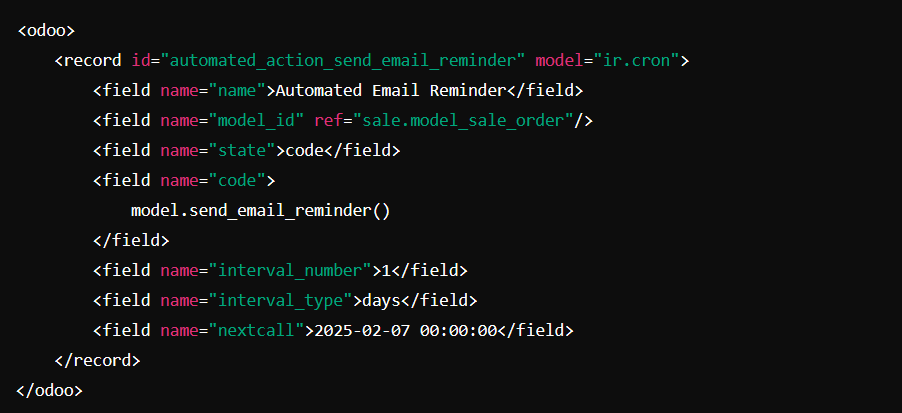
****

* **email\_template\_id:** Links to an existing email template.

**4. Automated Action**

Automated Actions are triggered based on certain conditions, such as when records are created or modified.

An automated action (or scheduled action) executes tasks at specific intervals or based on conditions. They are defined using the ir.cron model.



* **interval\_number:** Defines how often the action runs.
* **interval\_type:** Defines the interval type (minutes, hours, days, weeks, etc.).
* **code:** Python code executed when the action is triggered.

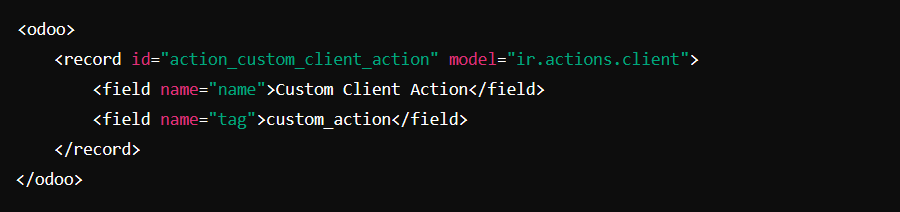
**5. Client Action**

Client Actions are written in JavaScript and are used for modifying the user interface and handling client-side logic.

A client action is used to execute JavaScript code or trigger a specific client-side behavior. It is useful for implementing custom UI interactions like **popups, notifications, or custom views**.

**Example: Client Action to modify views**

This action could modify how the form view looks based on specific criteria.



* **tag**: The tag is a JavaScript function that executes on the client side, triggering custom behavior (like opening a custom widget or altering the form view).
* **tag:** Specifies the client-side JavaScript action to execute.
* **params:** Passes additional parameters (optional) to the JavaScript action.

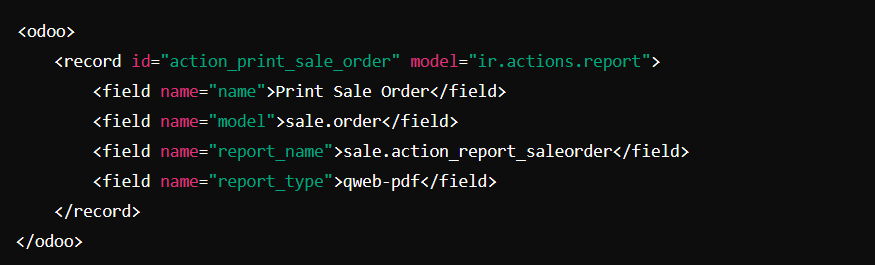
**6. Report Action**

Report Actions are used to generate printable reports (e.g., PDFs, Excel files).

A report action generates a report in a specific format **(PDF, XLSX, etc.)** and is often linked to a **button or menu**. Reports are created using QWeb templates.

**Example: Action to generate a PDF report**

This action triggers a report generation for a Sales Order.



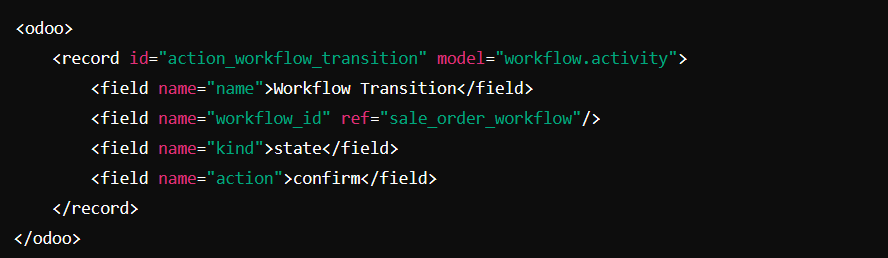
* **report\_name**: Defines the report template to use (sale.action\_report\_saleorder).
* **report\_type**: Specifies the type of report (e.g., qweb-pdf for PDF reports).

**7. Action Workflow**

Action Workflows are used for defining process steps and transitions (though workflows are deprecated in favor of automated actions in newer versions).

**Example: Action to trigger a workflow**

This action could transition a document (like a Sales Order) through a workflow stage.



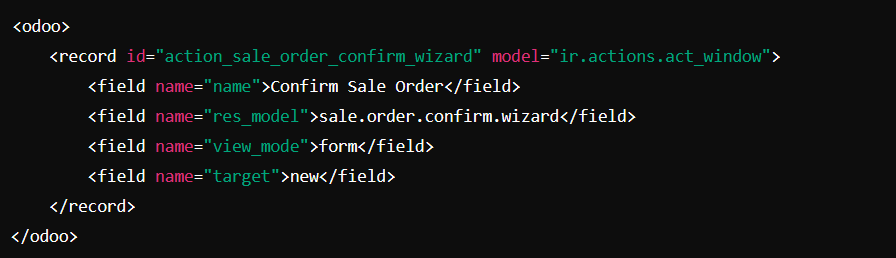
* **workflow\_id**: The workflow that manages the document.
* **action**: The specific transition to trigger.

**8. Wizard Action**

Wizards in Odoo are used for interacting with users through pop-up forms.

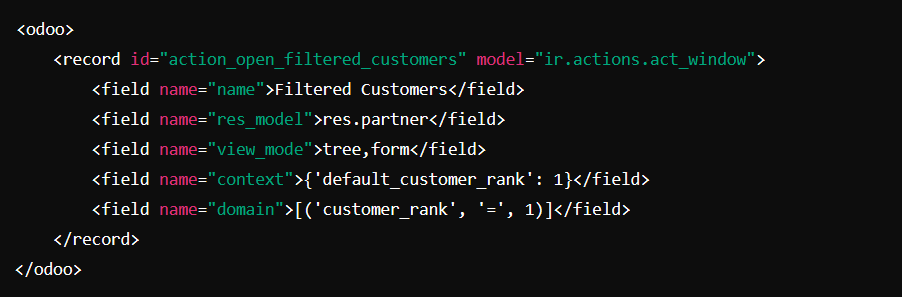
**Example: Action to open a wizard**

This action opens a wizard for confirming a Sales Order.



* **res\_model**: The model for the wizard (sale.order.confirm.wizard).
* **view\_mode**: Defines the view type (form for a form view).
* **target**: Specifies that the action opens a new window (new).

**9. Contextual Actions**  
Contextual actions modify behavior dynamically based on the context, such as user roles or record states. These are often linked with server actions or window actions.

****

* **context:** Passes default values or dynamic parameters.
* **domain:** Filters records based on conditions.

**Window Action**

* **Use when:** You need to navigate between views or open specific records in a list, form, or kanban view.
* **Example:** Clicking a menu item to open a list of customers or sales orders.

**Server Action**

* **Use when:** You need to execute server-side logic, such as sending emails, updating fields, or creating records based on certain conditions.
* **Example:** Automatically updating the status of a customer or sending a confirmation email.

**Client Action**

* **Use when:** You need to execute client-side actions like showing notifications, popups, or custom front-end behaviors (JavaScript).
* **Example:** Displaying a custom popup message when a user clicks a button.

**Report Action**

* **Use when:** You need to generate and display reports in formats like PDF or Excel.
* **Example:** Generating an invoice report or a sales summary.

**URL Action**

* **Use when:** You want to redirect the user to an external URL or an internal page.
* **Example:** Redirecting to an external documentation page or another Odoo module.

**Wizard Action**

* **Use when:** You need to get user input or perform a task in a wizard-like dialog. Wizards are temporary models for quick user interactions.
* **Example:** A wizard for bulk updating records or exporting data.

**Automated Action**

* **Use when:** You need to execute actions automatically at scheduled intervals or based on specific triggers (e.g., cron jobs).
* **Example:** Automatically sending reminder emails every day or updating records on a schedule.

**Contextual Actions**

* **Use when:** You want actions to behave dynamically based on context, such as user roles, record states, or other conditions.
* **Example:** Displaying specific buttons or fields depending on the user’s permissions.

**Summary Table: Ways to Call Each Action**

| **Action Type** | **Menu Item** | **Button (XML)** | **Python Code** | **JavaScript** | **Automated Action** | **OnChange** |
| --- | --- | --- | --- | --- | --- | --- |
| **ir.actions.act\_window** | ✅ Yes | ✅ Yes | ✅ Yes | ✅ Yes | ✅ Yes | ✅ Yes |
| **ir.actions.client** | ❌ No | ✅ Yes | ❌ No | ✅ Yes | ✅ Yes | ❌ No |
| **ir.actions.report** | ❌ No | ✅ Yes | ✅ Yes | ✅ Yes | ✅ Yes | ❌ No |
| **ir.actions.server** | ❌ No | ✅ Yes | ✅ Yes | ❌ No | ✅ Yes | ❌ No |
| **ir.actions.act\_url** | ❌ No | ✅ Yes | ✅ Yes | ✅ Yes | ✅ Yes | ❌ No |
| **ir.actions.todo** (deprecated) | ❌ No | ❌ No | ❌ No | ❌ No | ❌ No | ❌ No |