



[AutoGen](#)

S

[🏠](#) [Reference](#) [agentchat](#) [agent](#)

On this page

agentchat.agent

Agent Objects

```
class Agent()
```

(In preview) An abstract class for AI agent.

An agent can communicate with other agents and perform actions. Different agents can differ in what actions they perform in the `receive` method.

`__init__`

```
def __init__(name: str)
```

Arguments:

- `name str` - name of the agent.

`name`

```
@property
def name()
```

Get the name of the agent.

`send`

```
def send(message: Union[Dict, str],
         recipient: "Agent",
         request_reply: Optional[bool] = None)
```

(Abstract method) Send a message to another agent.

`a_send`

```
async def a_send(message: Union[Dict, str],
                 recipient: "Agent",
                 request_reply: Optional[bool] = None)
```

(Abstract async method) Send a message to another agent.

`receive`

```
def receive(message: Union[Dict, str],
            sender: "Agent",
            request_reply: Optional[bool] = None)
```

(Abstract method) Receive a message from another agent.

`a_receive`

```
async def a_receive(message: Union[Dict, str],
                    sender: "Agent",
                    request_reply: Optional[bool] = None)
```

(Abstract async method) Receive a message from another agent.

`reset`

```
def reset()
```

(Abstract method) Reset the agent.

generate_reply

```
def generate_reply(messages: Optional[List[Dict]] = None,
                   sender: Optional["Agent"] = None,
                   **kwargs) -> Union[str, Dict, None]
```

(Abstract method) Generate a reply based on the received messages.

Arguments:

- `messages` *list[dict]* - a list of messages received.
- `sender` - sender of an Agent instance.

Returns:

str or dict or None: the generated reply. If None, no reply is generated.

a_generate_reply

```
async def a_generate_reply(messages: Optional[List[Dict]] = None,
                           sender: Optional["Agent"] = None,
                           **kwargs) -> Union[str, Dict, None]
```

(Abstract async method) Generate a reply based on the received messages.

Arguments:

- `messages` *list[dict]* - a list of messages received.
- `sender` - sender of an Agent instance.

Returns:

str or dict or None: the generated reply. If None, no reply is generated.

 [Edit this page](#)

[Previous](#)
[« text_analyzer_agent](#)

[Next](#)
[assistant_agent »](#)

Community

[Discord](#) 

[Twitter](#) 