

Making a New Feature

How to Get Started Developing New Features

Adding a New Cog

1. Create a new Python file in the cogs/ directory, e.g. my_feature.py.
2. Define a class that inherits from commands.Cog and decorate it accordingly.
 - For example:

```
from discord.ext import commands

class MyFeatureCog(commands.Cog):
    def __init__(self, bot):
        self.bot = bot

    @commands.command()
    async def my_command(self, ctx):
        await ctx.send("Hello from my new feature!")

def setup(bot):
    bot.add_cog(MyFeatureCog(bot))
```

3. Import functions from `utils/helpers.py` in your cogs:

```
import utils.helpers as helpers
```

4. Use functions in `utils/helpers.py`:

```
variable = helpers.function(arguments)
# or,
helpers.function(arguments)
```

Best Practices

1. **Keep Cogs Focused:** Each cog should focus on a specific set of related commands or functionality to maintain clarity and organization.
 2. **Use Helper Functions:** Utilize `helpers.py` for any common functionality that may be used in multiple cogs.
 3. **Documentation:** Comment and document code in your cogs and helpers to make it easier for team members to understand the purpose and usage of code.
-

Revision #2

Created 18 May 2025 03:16:58 by Admin

Updated 18 May 2025 03:17:47 by Admin