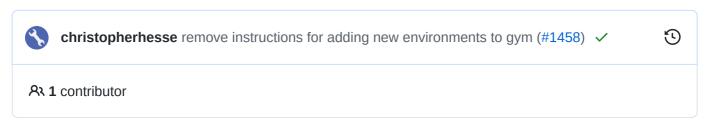
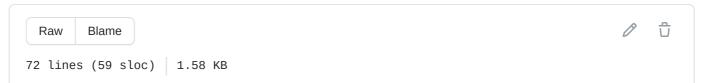


Code Issues 241 Pull requests 51 Actions Projects Wiki Security Insigl



gym / docs / creating-environments.md





How to create new environments for Gym

- Create a new repo called gym-foo, which should also be a PIP package.
- A good example is https://github.com/openai/gym-soccer.
- It should have at least the following files:

```
gym-foo/
README.md
setup.py
gym_foo/
   __init__.py
envs/
   __init__.py
foo_env.py
foo_extrahard_env.py
```

gym-foo/setup.py should have:

```
from setuptools import setup
setup(name='gym_foo',
    version='0.0.1',
```

```
install_requires=['gym'] # And any other dependencies foo needs
    )
gym-foo/gym_foo/__init__.py should have:
    from gym.envs.registration import register
    register(
        id='foo-v0',
        entry_point='gym_foo.envs:FooEnv',
    register(
        id='foo-extrahard-v0',
        entry_point='gym_foo.envs:FooExtraHardEnv',
    )
• gym-foo/gym_foo/envs/__init__.py should have:
    from gym_foo.envs.foo_env import FooEnv
    from gym_foo.envs.foo_extrahard_env import FooExtraHardEnv
• gym-foo/gym_foo/envs/foo_env.py should look something like:
    import gym
    from gym import error, spaces, utils
    from gym.utils import seeding
    class FooEnv(gym.Env):
      metadata = {'render.modes': ['human']}
```

```
from gym import error, spaces, utils
from gym.utils import seeding

class FooEnv(gym.Env):
    metadata = {'render.modes': ['human']}

    def __init__(self):
        ...
    def step(self, action):
        ...
    def reset(self):
        ...
    def render(self, mode='human'):
        ...
    def close(self):
        ...
```

 After you have installed your package with pip install -e gym-foo, you can create an instance of the environment with gym.make('gym_foo:foo-v0')