
IT314 Software Engineering

Lab 05: Static Analysis

ID: 202001139

Name: Poriya Neel Dharmeshbhai

Cloning repo with link: <https://github.com/shyam999/Django-ecommerce>

```
Command Prompt

C:\Users\student\Downloads>git clone https://github.com/shyam999/Django-ecommerce.git
Cloning into 'Django-ecommerce'...
remote: Enumerating objects: 363, done.
remote: Counting objects: 100% (1/1), done.
Receiving objects: 93% (338/363)ed 0 (delta 0), pack-reused 362
Receiving objects: 100% (363/363), 1.02 MiB | 3.18 MiB/s, done.
Resolving deltas: 100% (147/147), done.

C:\Users\student\Downloads>
```

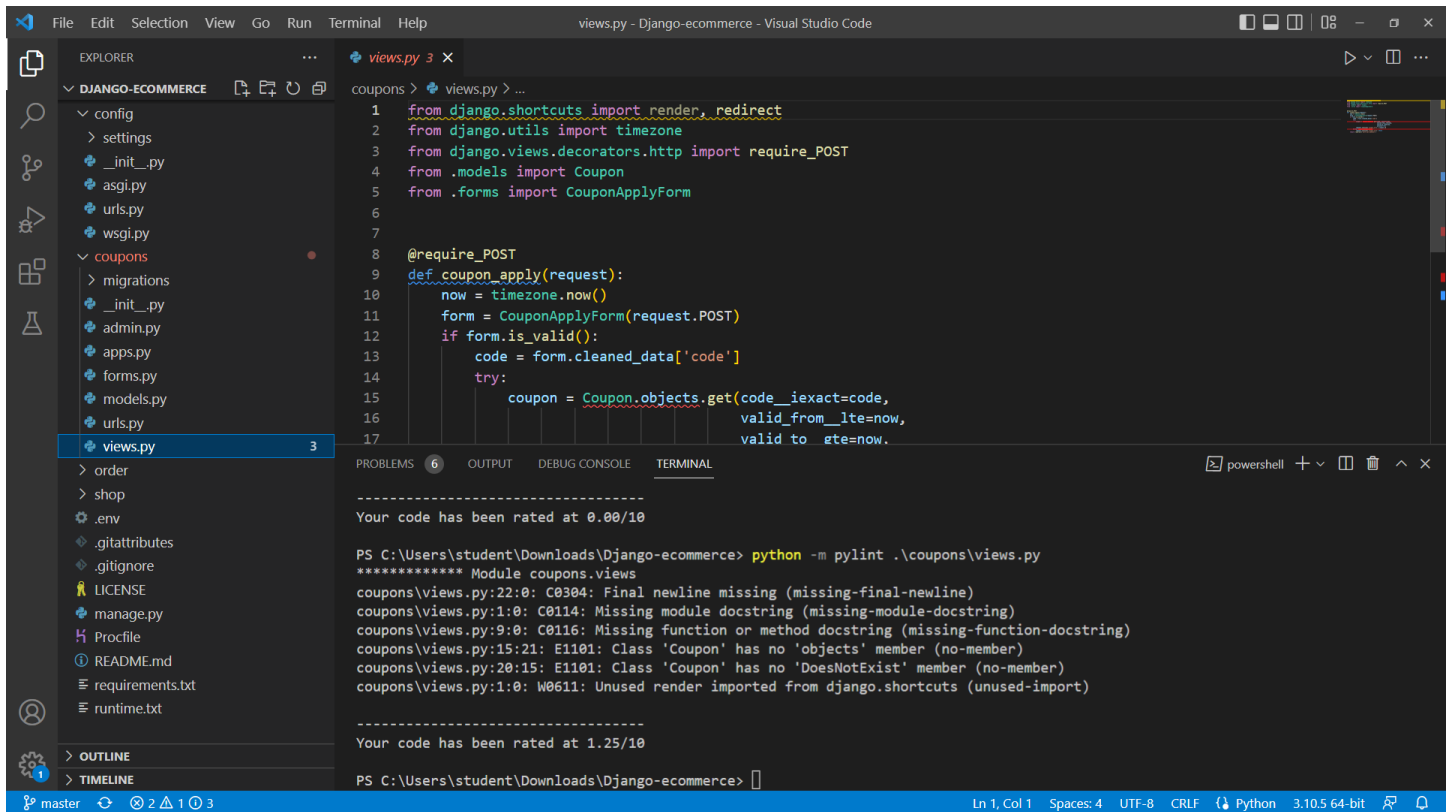
Installing pylint:

```
Command Prompt

C:\Users\student\Downloads\Django-ecommerce>python -m pip install pylint
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: pylint in c:\users\student\appdata\roaming\python\python310\site-packages (2.14.5)
Requirement already satisfied: astroid<2.12.0-dev0, >=2.11.6 in c:\users\student\appdata\roaming\python\python310\site-packages (from pylint) (2.11.6)
Requirement already satisfied: mccabe<0.8, >=0.6 in c:\users\student\appdata\roaming\python\python310\site-packages (from astroid<2.12.0-dev0, >=2.11.6) (0.7.0)
Requirement already satisfied: tomlkit>=0.10.1 in c:\users\student\appdata\roaming\python\python310\site-packages (from astroid<2.12.0-dev0, >=2.11.6) (0.11.4)
Requirement already satisfied: colorama>=0.4.5 in c:\users\student\appdata\roaming\python\python310\site-packages (from astroid<2.12.0-dev0, >=2.11.6) (0.4.5)
Requirement already satisfied: isort<6, >=4.2.5 in c:\users\student\appdata\roaming\python\python310\site-packages (from astroid<2.12.0-dev0, >=2.11.6) (5.10.1)
Requirement already satisfied: platformdirs>=2.2.0 in c:\users\student\appdata\roaming\python\python310\site-packages (from astroid<2.12.0-dev0, >=2.11.6) (2.6.0)
Requirement already satisfied: tomli>=1.1.0 in c:\users\student\appdata\roaming\python\python310\site-packages (from astroid<2.12.0-dev0, >=2.11.6) (2.0.1)
Requirement already satisfied: dill>=0.2 in c:\users\student\appdata\roaming\python\python310\site-packages (from pylint) (0.3.6)
Requirement already satisfied: setuptools>=20.0 in c:\program files\python310\lib\site-packages (from astroid<2.12.0-dev0, >=2.11.6) (57.0.0)
Requirement already satisfied: wrapt<2, >=1.11 in c:\users\student\appdata\roaming\python\python310\site-packages (from astroid<2.12.0-dev0, >=2.11.6) (1.14.1)
Requirement already satisfied: lazy-object-proxy>=1.4.0 in c:\users\student\appdata\roaming\python\python310\site-packages (from astroid<2.12.0-dev0, >=2.11.6) (1.7.1)

C:\Users\student\Downloads\Django-ecommerce>
```

Analyzing file: ./coupons/views.py



The screenshot shows the Visual Studio Code interface with the file `views.py` open in the editor. The file contains the following code:

```
1 from django.shortcuts import render, redirect
2 from django.utils import timezone
3 from django.views.decorators.http import require_POST
4 from .models import Coupon
5 from .forms import CouponApplyForm
6
7
8 @require_POST
9 def coupon_apply(request):
10     now = timezone.now()
11     form = CouponApplyForm(request.POST)
12     if form.is_valid():
13         code = form.cleaned_data['code']
14         try:
15             coupon = Coupon.objects.get(code__iexact=code,
16                                       valid_from__lte=now,
17                                       valid_to__gte=now.
```

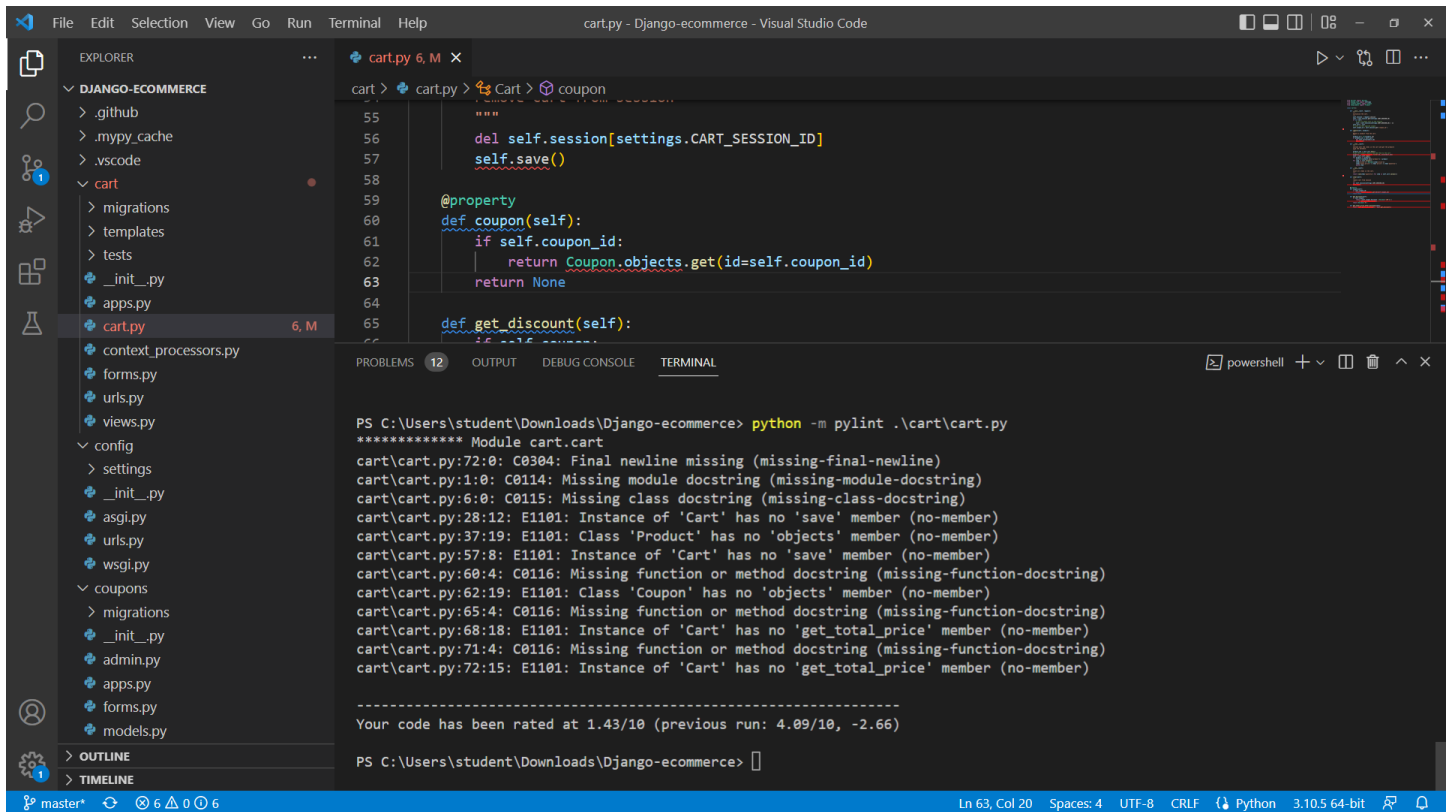
The terminal shows the output of the command `python -m pylint ./coupons/views.py`:

```
PS C:\Users\student\Downloads\Django-e-commerce> python -m pylint ./coupons/views.py
***** Module coupons.views
coupons\views.py:22:0: C0304: Final newline missing (missing-final-newline)
coupons\views.py:1:0: C0114: Missing module docstring (missing-module-docstring)
coupons\views.py:9:0: C0116: Missing function or method docstring (missing-function-docstring)
coupons\views.py:15:21: E1101: Class 'Coupon' has no 'objects' member (no-member)
coupons\views.py:20:15: E1101: Class 'Coupon' has no 'DoesNotExist' member (no-member)
coupons\views.py:1:0: W0611: Unused render imported from django.shortcuts (unused-import)
```

Analysis:

- **C0304 : Missing Module Error:** A module docstring is a string that appears at the beginning of a Python module and provides information about the module's purpose, contents, and usage. Pylint expects every module to have a docstring and raises an error if it's missing.
- **E1101 Class 'Coupon' has no 'objects' member:** The E1101 error in Pylint is a warning message that indicates an instance attribute has been accessed before it has been defined. This error typically occurs when an instance attribute is accessed within a class or method, but it has not been defined within the class or its parent classes.
- **W0611 Unused render imported from django.shortcuts:** The W0611 error in Pylint is a warning message that indicates an unused variable. This error is triggered when a variable is defined but not used anywhere in the code.

Analyzing ./cart/cart.py file:



The screenshot shows the Visual Studio Code interface with the file explorer on the left displaying the project structure. The main editor shows the `cart.py` file with the following code:

```
55
56
57     del self.session[settings.CART_SESSION_ID]
58     self.save()
59
60 @property
61 def coupon(self):
62     if self.coupon_id:
63         return Coupon.objects.get(id=self.coupon_id)
64     return None
65
66 def get_discount(self):
67     if self.coupon:
```

The terminal at the bottom shows the output of the command `python -m pylint ./cart/cart.py`:

```
PS C:\Users\student\Downloads\Django-e-commerce> python -m pylint ./cart/cart.py
***** Module cart.cart
cart\cart.py:72:0: C0304: Final newline missing (missing-final-newline)
cart\cart.py:1:0: C0114: Missing module docstring (missing-module-docstring)
cart\cart.py:6:0: C0115: Missing class docstring (missing-class-docstring)
cart\cart.py:28:12: E1101: Instance of 'Cart' has no 'save' member (no-member)
cart\cart.py:37:19: E1101: Class 'Product' has no 'objects' member (no-member)
cart\cart.py:57:8: E1101: Instance of 'Cart' has no 'save' member (no-member)
cart\cart.py:60:4: C0116: Missing function or method docstring (missing-function-docstring)
cart\cart.py:62:19: E1101: Class 'Coupon' has no 'objects' member (no-member)
cart\cart.py:65:4: C0116: Missing function or method docstring (missing-function-docstring)
cart\cart.py:68:18: E1101: Instance of 'Cart' has no 'get_total_price' member (no-member)
cart\cart.py:71:4: C0116: Missing function or method docstring (missing-function-docstring)
cart\cart.py:72:15: E1101: Instance of 'Cart' has no 'get_total_price' member (no-member)

Your code has been rated at 1.43/10 (previous run: 4.09/10, -2.66)
```

Analysis:

- **C0114 : Missing Module Error:** A module docstring is a string that appears at the beginning of a Python module and provides information about the module's purpose, contents, and usage. Pylint expects every module to have a docstring and raises an error if it's missing.
- **E1101 : Instance of 'Cart' has no 'save' member:** the error message "Instance of 'Cart' has no 'save' member" indicates that you are trying to call a method named `save` on an instance of the `Cart` class, but that method does not exist on the `Cart` class.
- **C0116 : Missing function docstring:** This error message indicates that Pylint has detected a missing docstring in your function or method. To resolve this error, you should add a docstring to your function or method.
- **E1101 : Instance of 'Cart' has no 'get_total_price' member:** the error message "Instance of 'Cart' has no 'get_total_price' member" indicates that you are trying to call a method named `get_total_price` on an instance of the `Cart` class, but that method does not exist on the `Cart` class.

Analyzing file ./shop/views.py

```
shop > views.py > ...
12 # '''Return five latest products
13 #
14 # return Product.objects.filter(created_lte=timezone.now()
15 # ).order_by('-created')[:5]
16
17
18
19 def product_list(request, category_slug=None):
20     category = None
21     categories = Category.objects.all()
22     products = Product.objects.filter(available=True)
23     if category_slug:
24         category = get_object_or_404(Category, slug=category_slug)
25         products = products.filter(category=category)
```

```
PS C:\Users\student\Downloads\Django-e-commerce> python -m pylint ./shop/views.py
***** Module shop.views
shop\views.py:63:58: C0303: Trailing whitespace (trailing-whitespace)
shop\views.py:65:0: C0304: Final newline missing (missing-final-newline)
shop\views.py:1:0: C0114: Missing module docstring (missing-module-docstring)
shop\views.py:19:0: C0116: Missing function or method docstring (missing-function-docstring)
shop\views.py:21:17: E1101: Class 'Category' has no 'objects' member (no-member)
shop\views.py:22:15: E1101: Class 'Product' has no 'objects' member (no-member)
shop\views.py:48:0: C0116: Missing function or method docstring (missing-function-docstring)
shop\views.py:48:28: C0103: Argument name "id" doesn't conform to snake_case naming style (invalid-name)
shop\views.py:48:28: W0622: Redefining built-in 'id' (redefined-builtin)

-----
Your code has been rated at 0.00/10

PS C:\Users\student\Downloads\Django-e-commerce>
```

Analysis:

- **C0303: Trailing whitespace:** This error message indicates that Pylint has detected one or more lines of code that end with trailing whitespace. To resolve this error, you should remove the trailing whitespace from the affected lines of code.
- **C0114: Missing Module Error:** A module docstring is a string that appears at the beginning of a Python module and provides information about the module's purpose, contents, and usage. Pylint expects every module to have a docstring and raises an error if it's missing.
- **C0103: Argument name 'id' doesn't conform to snake_case styling:** This error shows that proper variable naming conventions is not used. Here pylint suggests that we should use snake_case convention for naming variables.
- **W0622: Redefining built-in 'id':** Used when a variable or function override a built-in. This means that the 'id' variable is redefined again somewhere inside code.

Analyzing file : ./order/models.py

The screenshot shows the Visual Studio Code interface with the file `models.py` open in the editor. The file contains the following code:

```
order > models.py > OrderItem
22     return sum(item.get_cost() for item in self.items.all())
23
24 class OrderItem(models.Model):
25     order = models.ForeignKey(Order, related_name='items', on_delete=models.CASCADE)
26     product = models.ForeignKey(Product, related_name='order_items', on_delete=models.CASCADE)
27
28     def __str__(self):
29         return '{}'.format(self.id)
30
31     def get_cost(self):
32         return self.price * self.quantity
33
```

The bottom panel shows the Pylint analysis results in the terminal:

```
PS C:\Users\student\Downloads\Django-e-commerce> python -m pylint ./order/models.py
***** Module order.models
order\models.py:1:0: C0114: Missing module docstring (missing-module-docstring)
order\models.py:4:0: C0115: Missing class docstring (missing-class-docstring)
order\models.py:15:4: C0115: Missing class docstring (missing-class-docstring)
order\models.py:15:4: R0903: Too few public methods (0/2) (too-few-public-methods)
order\models.py:19:15: C0209: Formatting a regular string which could be a f-string (consider-using-f-string)
order\models.py:19:33: E1101: Instance of 'Order' has no 'id' member (no-member)
order\models.py:21:4: C0116: Missing function or method docstring (missing-function-docstring)
order\models.py:22:47: E1101: Instance of 'Order' has no 'items' member (no-member)
order\models.py:24:0: C0115: Missing class docstring (missing-class-docstring)
order\models.py:29:15: C0209: Formatting a regular string which could be a f-string (consider-using-f-string)
order\models.py:29:27: E1101: Instance of 'OrderItem' has no 'id' member (no-member)
order\models.py:31:4: C0116: Missing function or method docstring (missing-function-docstring)
order\models.py:32:15: E1101: Instance of 'OrderItem' has no 'price' member (no-member)
order\models.py:32:28: E1101: Instance of 'OrderItem' has no 'quantity' member (no-member)

-----
Your code has been rated at 0.00/10 (previous run: 0.00/10, +0.00)

PS C:\Users\student\Downloads\Django-e-commerce>
```

Analysis:

- **C0303 : Trailing whitespace:** This error message indicates that Pylint has detected one or more lines of code that end with trailing whitespace. To resolve this error, you should remove the trailing whitespace from the affected lines of code.
- **R0903 : Too few public methods:** The error basically says that classes aren't meant to just store data, as you're basically treating the class as a dictionary.
- **E1101 : Instance of 'OrderItem' has no 'price' member:** The E1101 error is a type of error that occurs in Python when a module or class member is accessed that does not exist.

Analyzing file: ./shop/models.py

The screenshot shows the Visual Studio Code interface with the Django E-commerce project open. The Explorer pane on the left shows the project structure, including the 'shop' directory. The main editor displays the 'models.py' file, which contains a 'Meta' class and two methods: 'get_absolute_url' and '__str__'. The 'PROBLEMS' pane at the bottom shows a list of errors and warnings detected by pylint. The errors include missing docstrings, too few public methods, and an instance of 'Product' with no 'id' member. The 'TERMINAL' pane shows the command 'python -m pylint ./shop/models.py' and the resulting output, which includes the same list of errors and warnings. The status bar at the bottom indicates the current file is 'models.py' at line 14, column 25, with 4 spaces, UTF-8 encoding, and CRLF line endings.

```
shop > models.py > Category > _str_
31
32 class Meta:
33     ordering = ('name',)
34     index_together = (('id', 'slug'),)
35
36 def __str__(self):
37     return self.name
38
39 def get_absolute_url(self):
40     return reverse('shop:product_detail', args=[self.id, self.slug])
41
```

PROBLEMS 10 OUTPUT DEBUG CONSOLE TERMINAL

Your code has been rated at 3.00/10

```
PS C:\Users\student\Downloads\Django-e-commerce> python -m pylint ./shop/models.py
***** Module shop.models
shop\models.py:1:0: C0114: Missing module docstring (missing-module-docstring)
shop\models.py:4:0: C0115: Missing class docstring (missing-class-docstring)
shop\models.py:8:4: C0115: Missing class docstring (missing-class-docstring)
shop\models.py:8:4: R0903: Too few public methods (0/2) (too-few-public-methods)
shop\models.py:13:4: E0307: __str__ does not return str (invalid-str-returned)
shop\models.py:16:4: C0116: Missing function or method docstring (missing-function-docstring)
shop\models.py:20:0: C0115: Missing class docstring (missing-class-docstring)
shop\models.py:32:4: C0115: Missing class docstring (missing-class-docstring)
shop\models.py:32:4: R0903: Too few public methods (0/2) (too-few-public-methods)
shop\models.py:36:4: E0307: __str__ does not return str (invalid-str-returned)
shop\models.py:39:4: C0116: Missing function or method docstring (missing-function-docstring)
shop\models.py:40:52: E1101: Instance of 'Product' has no 'id' member (no-member)

Your code has been rated at 2.00/10 (previous run: 2.00/10, +0.00)

PS C:\Users\student\Downloads\Django-e-commerce>
```

Analysis:

- **C0303: Trailing whitespace:** This error message indicates that Pylint has detected one or more lines of code that end with trailing whitespace. To resolve this error, you should remove the trailing whitespace from the affected lines of code.
- **E1101: Instance of 'Product' has no 'id' member:** This error message indicates that you are trying to access an attribute called `id` on an instance of a class called `Product`, but the `id` attribute is not defined for that class.
- **R0903: Too few public methods:** This warning message indicates that a class has too few public methods, which could suggest a design issue or a potential problem with the class's functionality.
- **E0307: __str__ does not return str:** If the `__str__` method of a class does not return a string, it can lead to errors when the method is called. For example, if you try to print an object that has a faulty `__str__` method, you may get unexpected output or an error message.