Django\_apis

Here is the URL configuration for the "Product Viewing" use case:  
  
```  
urlpatterns = [  
 #... other URLs...  
 path('products/', ProductListView.as\_view(), name='product\_list'),  
 path('products/<int:pk>/', ProductDetailView.as\_view(), name='product\_detail'),  
]  
```  
  
Explanation: This URL pattern will map the `/products/` route to the `ProductListView`, which will display a list of products. The `<int:pk>` part is a parameter that will capture an integer value from the URL, which can then be used by the view to retrieve the corresponding product detail.  
  
Note: Make sure you have already defined your views (`ProductListView` and `ProductDetailView`) before using them in this URL pattern. If they haven't been defined yet, please define them first and then try running this URL pattern.

Here is the URL configuration for the "Product Viewing" use case:  
  
```  
urlpatterns = [  
 #... other URLs...  
 path('products/', ProductListView.as\_view(), name='product\_list'),  
 path('products/<int:pk>/', ProductDetailView.as\_view(), name='product\_detail'),  
]  
```  
  
Explanation: This URL pattern will map the `/products/` route to the `ProductListView`, which will display a list of products. The `<int:pk>` part is a parameter that will capture an integer value from the URL, which can then be used by the view to retrieve the corresponding product detail.  
  
Note: Make sure you have already defined your views (`ProductListView` and `ProductDetailView`) before using them in this URL pattern. If they haven't been defined yet, please define them first and then try running this URL pattern.

Here is the URL configuration for the "Product Viewing" use case:  
  
```  
urlpatterns = [  
 #... other URLs...  
 path('products/', ProductListView.as\_view(), name='product\_list'),  
 path('products/<int:pk>/', ProductDetailView.as\_view(), name='product\_detail'),  
]  
```  
  
Explanation: This URL pattern will map the `/products/` route to the `ProductListView`, which will display a list of products. The `<int:pk>` part is a parameter that will capture an integer value from the URL, which can then be used by the view to retrieve the corresponding product detail.  
  
Note: Make sure you have already defined your views (`ProductListView` and `ProductDetailView`) before using them in this URL pattern. If they haven't been defined yet, please define them first and then try running this URL pattern.

Here is the URL configuration for the "Product Viewing" use case:  
  
```  
urlpatterns = [  
 #... other URLs...  
 path('products/', ProductListView.as\_view(), name='product\_list'),  
 path('products/<int:pk>/', ProductDetailView.as\_view(), name='product\_detail'),  
]  
```  
  
Explanation: This URL pattern will map the `/products/` route to the `ProductListView`, which will display a list of products. The `<int:pk>` part is a parameter that will capture an integer value from the URL, which can then be used by the view to retrieve the corresponding product detail.  
  
Note: Make sure you have already defined your views (`ProductListView` and `ProductDetailView`) before using them in this URL pattern. If they haven't been defined yet, please define them first and then try running this URL pattern.

Here is the URL configuration for the "Product Viewing" use case:  
  
```  
urlpatterns = [  
 #... other URLs...  
 path('products/', ProductListView.as\_view(), name='product\_list'),  
 path('products/<int:pk>/', ProductDetailView.as\_view(), name='product\_detail'),  
]  
```  
  
Explanation: This URL pattern will map the `/products/` route to the `ProductListView`, which will display a list of products. The `<int:pk>` part is a parameter that will capture an integer value from the URL, which can then be used by the view to retrieve the corresponding product detail.  
  
Note: Make sure you have already defined your views (`ProductListView` and `ProductDetailView`) before using them in this URL pattern. If they haven't been defined yet, please define them first and then try running this URL pattern.