These are the commands for the follow-along exercises in Lesson 2.4.

2.402: Using the ORM in views.py

Code urls.py

```
from django.urls import include, path

    path('', include('genedata.urls')),
```

Code genedata/urls.py

```

from django.urls import include, path
from . import views

urlpatterns = [
    path('', views.index, name='index'),
]
```

Code genedata/views.py

```
def index(request):

    response = "Hello"
    return render(request, 'genedata/index.html', {'message': response})
```

Code index.html

```

<html>
  <head>
  </head>
  <body>
    <p>{{ message }}</p>
  </body>
</html>
```

Code genedata/views.py

```

from .models import *
def index(request):
    genes = Gene.objects.all()
    return render(request, 'genedata/index.html', {'genes': genes})
```

Code index.html

```
<html>
  <head>
  </head>
  <body>
    <h1>D.Bucha Genes</h1>
    <table>
      <tr><th>Gene ID</th></tr>

      {% for gene in genes %}
        <tr><td>{{gene.gene_id}}</td></tr>
      {% endfor %}
    </table>
  </body>
</html>

<tr><td>{{gene}}</td></tr>

        <tr><td><a href="/gene/{{gene.pk}}" >{{gene}}</a></td></tr>
```

Code genedata/urls.py

```
urlpatterns = [
    path('', views.index, name='index'),
    path('gene/<int:pk>', views.gene, name='gene'),
]
```

Code genedata/views.py

```
def gene(request, pk):
    gene = Gene.objects.get(pk=pk)
    return render(request, 'genedata/gene.html', {'gene': gene})
```

Code gene.html

```
<html>
  <head>

  </head>
  <body>
    <h1>{{gene}}</h1>
    <table>
      <tr><th>Key</th><th>Value</th></tr>
        <tr><td>Entity: </td><td>{{gene.entity}}</td></tr>
        <tr><td>Start: </td><td>{{gene.start}}</td></tr>
        <tr><td>Stop: </td><td>{{gene.stop}}</td></tr>
        <tr><td>Sense: </td><td>{{gene.sense}}</td></tr>
        <tr><td>Start Codon: </td><td>{{gene.start_codon}}</td></tr>
    </table>
  </body>
</html>
```

2.404: Joins, filters and chaining commands

Code gene.html

```
        {% with seq=gene.sequencing %}
        <tr><td>Sequencing Factory: 
</td><td>{{seq.sequencing_factory}}</td></tr>
        <tr><td>Factory Location: </td><td>{{seq.factory_location}}</td></tr>
        {% endwith %}
        {% with ec=gene.ec %}
        <tr><td>EC Name: </td><td>{{ec.ec_name}}</td></tr>
        {% endwith %}
       {% with links=gene.geneattributelink_set.all %}
         {% for link in links %}

<tr><td>{{link.attribute.key}}</td><td>{{link.attribute.value}}</td></tr>
         {% endfor %}
       {% endwith %}
```

Code index.html

```
<h2>Select gene location</h2>
<a href="/list/Chromosome">Chromosome</a> OR <a href="/list/Plasmid">Plasmid</a>
```

Code genedata/urls.py

```
    path('list/<str:type>', views.list, name='list'),
```

Code views.py

```
def list(request, type):
    genes = Gene.objects.filter(entity__exact=type)
    return render(request, 'genedata/list.html', {'genes': genes, 'type': type})
```

Code list.html

```
<html>
 <head>
 </head>
 <body>
 <h1>Filtered List: {{type}}</h1>
 <table>
 <tr><th>Gene ID</th></tr>
 {% for gene in genes %}
 <tr><td><a href="/gene/{{gene.pk}}">{{gene}}</a></td></tr>
 {% endfor %}
 </table>
  </body>
</html>
```

Code index.html

```
<h2>Show Positive Chromosome</h2>
<a href="/poslist/">Show This List</a>
```

Code genedata/urls.py

```
    path('poslist/', views.poslist, name='poslist'),
```

Code views.py

```
def poslist(request):
    genes =
Gene.objects.filter(entity__exact='Chromosome').filter(sense__startswith='+')
    return render(request, 'genedata/list.html', {'genes': genes, 'type':
'PosList'})
```

2.406: Deleting and updating records

Code models.py

```
access = models.IntegerField(null=False, blank=False, default=0)
```

Code views.py

```
def gene(request, pk):
   gene = Gene.objects.get(pk=pk)
   gene.access += 1
   print("Gene record:", pk, "access count:", str(gene.access))
   gene.save()
   return render(request, 'genedata/gene.html', {'gene': gene})
```

code gene.html

```
<a href="/delete/{{gene.pk}}">DELETE RECORD</a>
```

code genedata/urls.py

```
   path('delete/<int:pk>', views.delete, name='delete'),
```

Code views.py

```
from django.http import HttpResponseRedirect

def delete(request, pk):
   GeneAttributeLink.objects.filter(gene_id=pk).delete()
   Gene.objects.filter(pk=pk).delete()
   return HttpResponseRedirect("/")
```