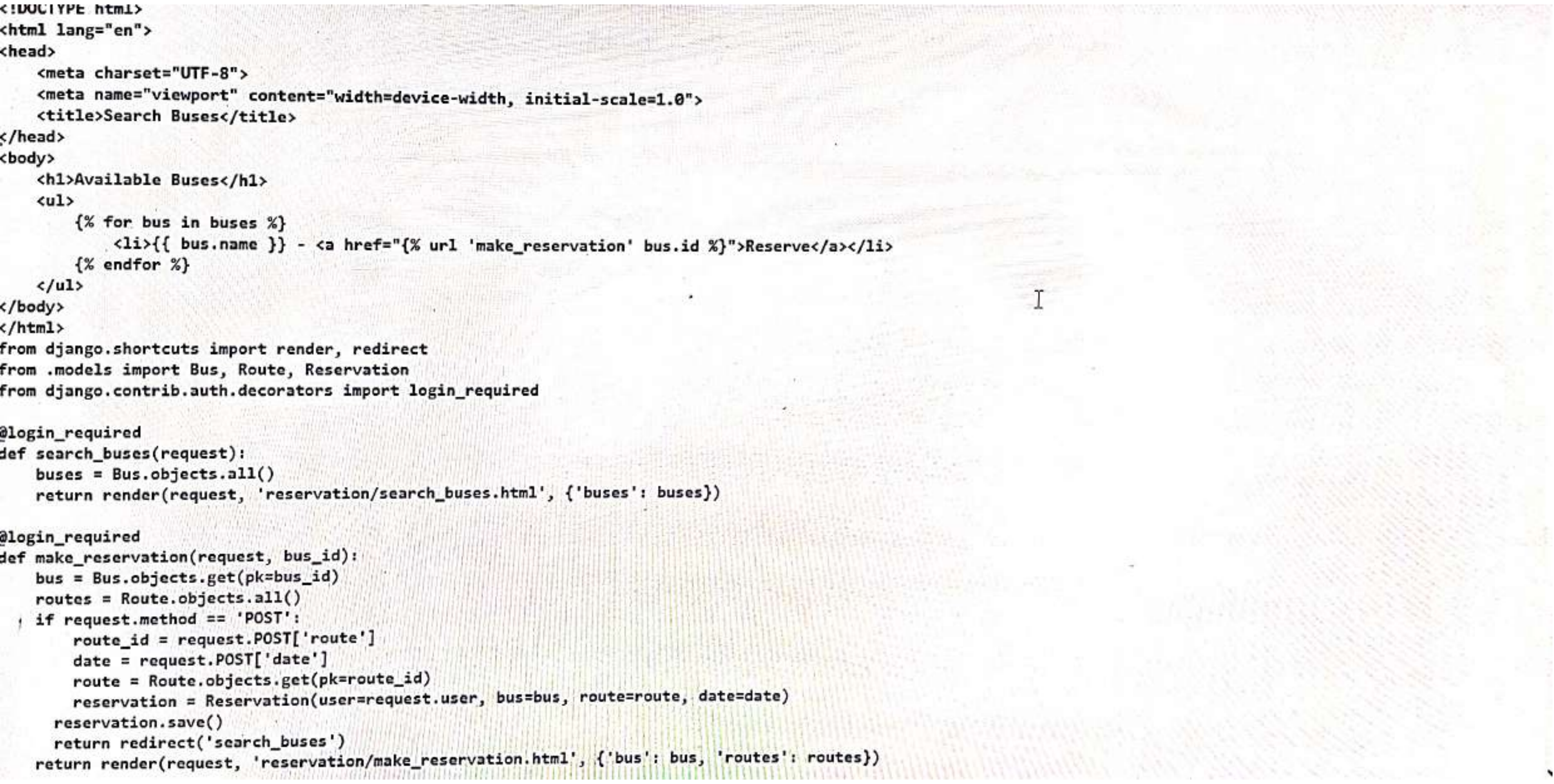
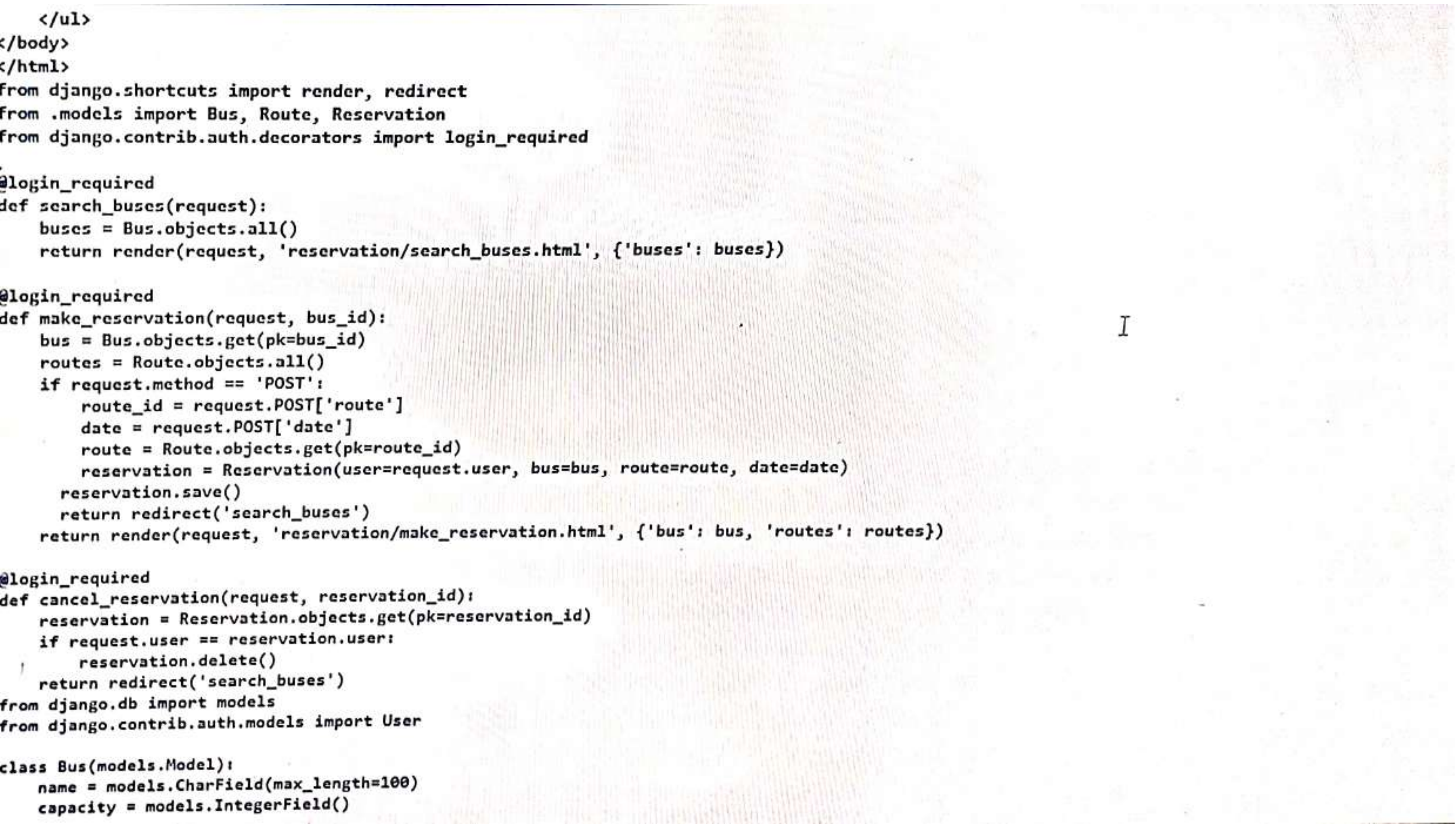
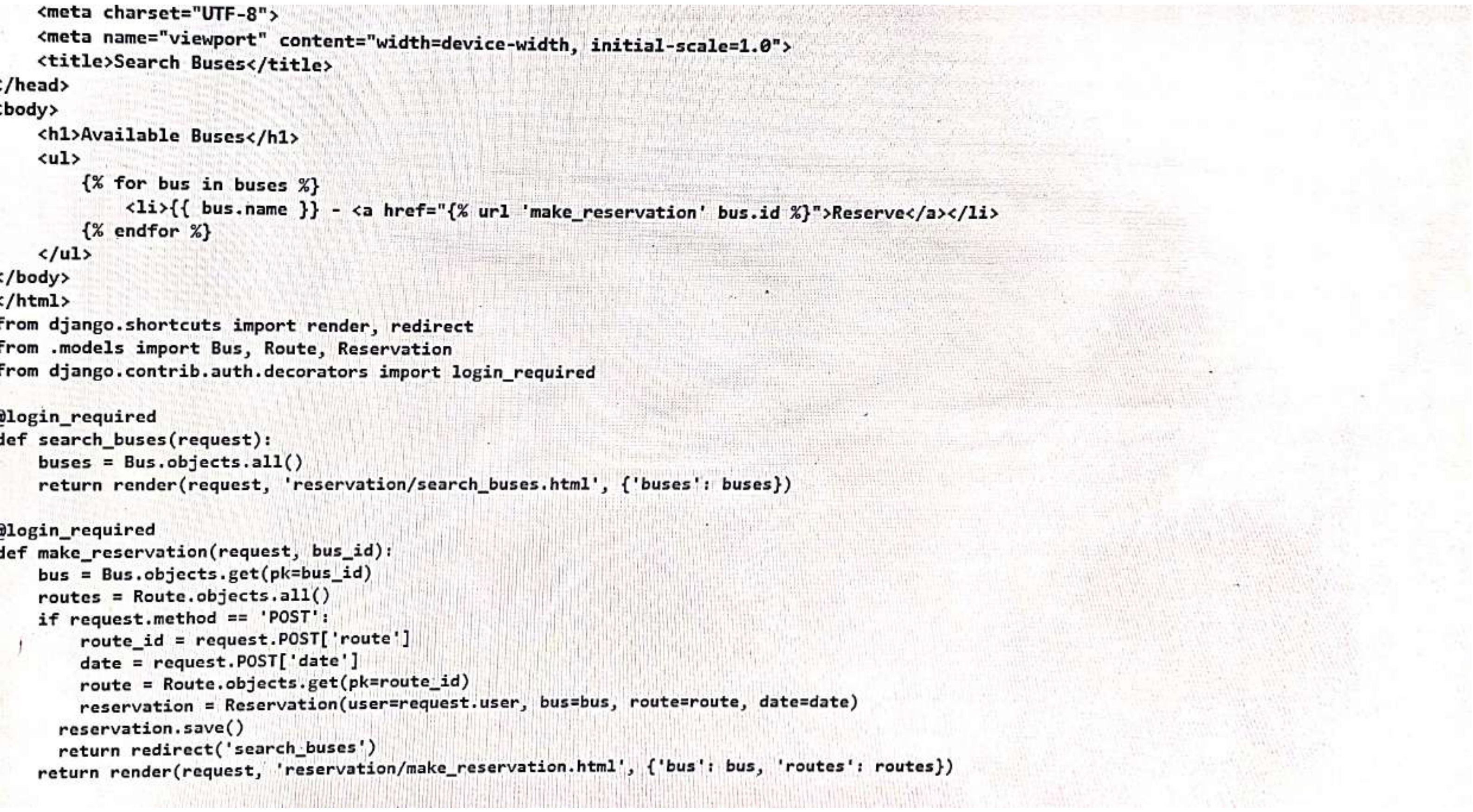


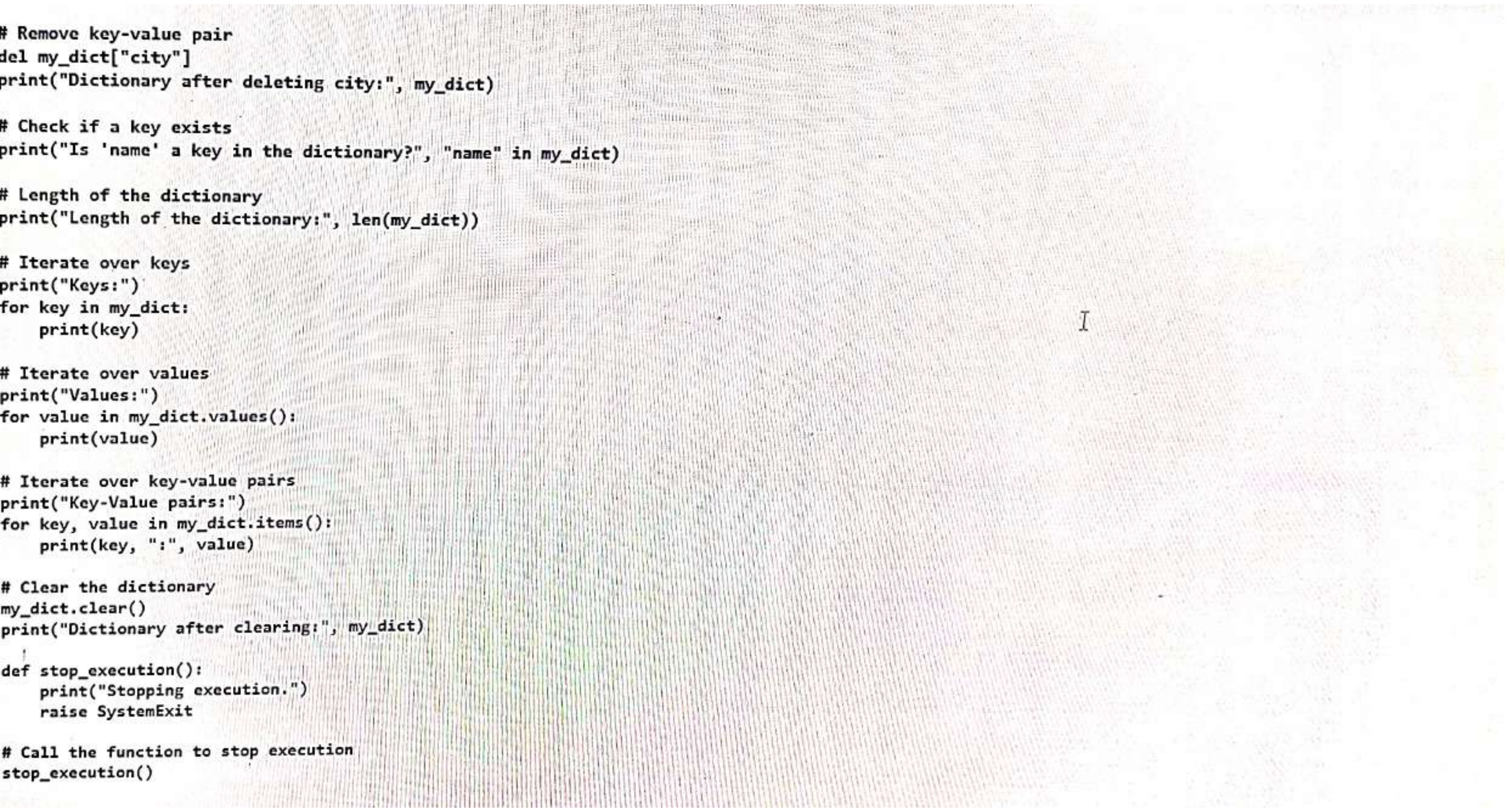
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Search Buses</title>
</head>
<body>
    <h1>Available Buses</h1>
    <ul>
        {% for bus in buses %}
            <li>{{ bus.name }} - <a href="{% url 'make_reservation' bus.id %}">Reserve</a></li>
        {% endfor %}
    </ul>
</body>
</html>
from django.shortcuts import render, redirect
from .models import Bus, Route, Reservation
from django.contrib.auth.decorators import login_required

@login_required
def search_buses(request):
    buses = Bus.objects.all()
    return render(request, 'reservation/search_buses.html', {'buses': buses})

@login_required
def make_reservation(request, bus_id):
    bus = Bus.objects.get(pk=bus_id)
    routes = Route.objects.all()
    if request.method == 'POST':
        route_id = request.POST['route']
        date = request.POST['date']
        route = Route.objects.get(pk=route_id)
        reservation = Reservation(user=request.user, bus=bus, route=route, date=date)
      reservation.save()
      return redirect('search_buses')
    return render(request, 'reservation/make_reservation.html', {'bus': bus, 'routes': routes})
```



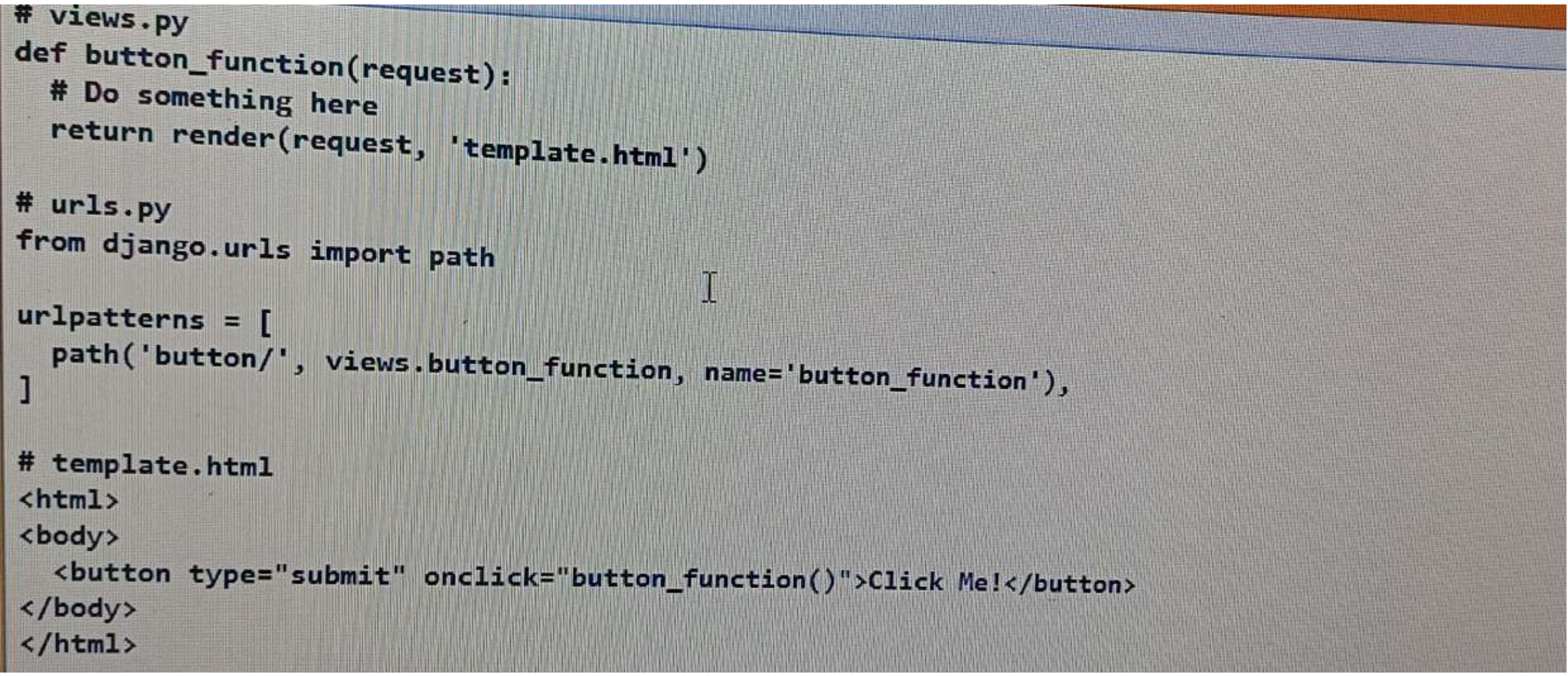
```
    </ul>
</body>
</html>
from django.shortcuts import render, redirect
from .models import Bus, Route, Reservation
from django.contrib.auth.decorators import login_required

@login_required
def search_buses(request):
    buses = Bus.objects.all()
    return render(request, 'reservation/search_buses.html', {'buses': buses})

@login_required
def make_reservation(request, bus_id):
    bus = Bus.objects.get(pk=bus_id)
    routes = Route.objects.all()
    if request.method == 'POST':
        route_id = request.POST['route']
        date = request.POST['date']
        route = Route.objects.get(pk=route_id)
        reservation = Reservation(user=request.user, bus=bus, route=route, date=date)
      reservation.save()
      return redirect('search_buses')
    return render(request, 'reservation/make_reservation.html', {'bus': bus, 'routes': routes})

@login_required
def cancel_reservation(request, reservation_id):
    reservation = Reservation.objects.get(pk=reservation_id)
    if request.user == reservation.user:
        reservation.delete()
    return redirect('search_buses')
from django.db import models
from django.contrib.auth.models import User

class Bus(models.Model):
    name = models.CharField(max_length=100)
    capacity = models.IntegerField()
```



```
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Search Buses</title>
</head>
<body>
    <h1>Available Buses</h1>
    <ul>
        {% for bus in buses %}
            <li>{{ bus.name }} - <a href="{% url 'make_reservation' bus.id %}">Reserve</a></li>
        {% endfor %}
    </ul>
</body>
</html>
from django.shortcuts import render, redirect
from .models import Bus, Route, Reservation
from django.contrib.auth.decorators import login_required

@login_required
def search_buses(request):
    buses = Bus.objects.all()
    return render(request, 'reservation/search_buses.html', {'buses': buses})

@login_required
def make_reservation(request, bus_id):
    bus = Bus.objects.get(pk=bus_id)
    routes = Route.objects.all()
    if request.method == 'POST':
        route_id = request.POST['route']
        date = request.POST['date']
        route = Route.objects.get(pk=route_id)
        reservation = Reservation(user=request.user, bus=bus, route=route, date=date)
      reservation.save()
      return redirect('search_buses')
    return render(request, 'reservation/make_reservation.html', {'bus': bus, 'routes': routes})
```



```
# Remove key-value pair
del my_dict["city"]
print("Dictionary after deleting city:", my_dict)

# Check if a key exists
print("Is 'name' a key in the dictionary?", "name" in my_dict)

# Length of the dictionary
print("Length of the dictionary:", len(my_dict))

# Iterate over keys
print("Keys:")
for key in my_dict:
    print(key)

# Iterate over values
print("Values:")
for value in my_dict.values():
    print(value)

# Iterate over key-value pairs
print("Key-Value pairs:")
for key, value in my_dict.items():
    print(key, ":", value)

# Clear the dictionary
my_dict.clear()
print("Dictionary after clearing:", my_dict)

def stop_execution():
    print("Stopping execution.")
    raise SystemExit

# Call the function to stop execution
stop_execution()
```



```
# views.py
def button_function(request):
  # Do something here
  return render(request, 'template.html')

# urls.py
from django.urls import path

urlpatterns = [
  path('button/', views.button_function, name='button_function'),
]

# template.html
<html>
<body>
  <button type="submit" onclick="button_function()">Click Me!</button>
</body>
</html>
```