Question 1: Are Django signals executed synchronously or asynchronously by default?

By default, Django signals are executed synchronously. This means that the signal handler is executed in the same thread, immediately after the signal is sent, and the main thread will wait for the signal handler to finish before proceeding.

```
import time
from django.db.models.signals import post_save
from django.dispatch import receiver
from django.contrib.auth.models import User

@receiver(post_save, sender=User)
def signal_handler(sender, instance, created, **kwargs):
    print("Signal handler started")
    time.sleep(5)  # Simulate a long-running task
    print("Signal handler finished")

# Simulating saving a user
user = User(username='test_user')
user.save()
print("User saved")
```

Question 2: Do Django signals run in the same thread as the caller?

Yes, by default, Django signals run in the **same thread** as the caller.

```
import threading
from django.db.models.signals import post_save
from django.dispatch import receiver
```

```
from django.contrib.auth.models import User

@receiver(post_save, sender=User)
def signal_handler(sender, instance, created, **kwargs):
    print(f"Signal handler running in thread: {threading.current_thread().name}")

# Simulating saving a user
print(f"Main thread: {threading.current_thread().name}")
user = User(username='test_user')
user.save()
```

Question 3: By default, do Django signals run in the same database transaction as the caller?

Yes, Django signals run in the same database transaction as the caller when they are triggered inside a database operation.

```
from django.db import transaction
from django.db.models.signals import post_save
from django.dispatch import receiver
from django.contrib.auth.models import User

@receiver(post_save, sender=User)
def signal_handler(sender, instance, created, **kwargs):
    print("Signal handler executed")

# Simulating a database transaction
with transaction.atomic():
    print("Transaction started")
    user = User(username='test_user')
```

```
    user.save()
    print("Transaction still active")

print("Transaction complete")
```