Creating a MassTransit state machine

Script start

As we said before, we will orchestrate our purchase saga via a MassTransit state machine. This state machine will keep track of the current state of the process and will drive the process from start to finish.

In Trading repo

1. dotnet add package MassTransit

2. Create the StateMachines directory

3. Add PurchaseState.cs:

```
namespace Play.Trading.Service.StateMachines
{
   public class PurchaseState : SagaStateMachineInstance, ISagaVersion
   {
      public Guid CorrelationId { get; set; }
      public string CurrentState { get; set; }
      public Guid UserId { get; set; }
      public Guid ItemId { get; set; }
      public int Quantity { get; set; }
      public DateTimeOffset Received { get; set; }
      public decimal? PurchaseTotal { get; set; }
      public DateTimeOffset LastUpdated { get; set; }
      public string ErrorMessage { get; set; }
      public int Version { get; set; }
   }
}
```

4. Add PurchaseStateMachine.cs:

```
namespace Play.Trading.Service.StateMachines
{
   public class PurchaseStateMachine : MassTransitStateMachine<PurchaseState>
   {
      public State Accepted { get; }
      public State ItemsGranted { get; }
      public State Completed { get; }
      public State Faulted { get; }

      public PurchaseStateMachine()
```

```
        {
            InstanceState(state => state.CurrentState);
        }
    }
}
```

In the next lesson we will see how to initialize the state machine.