

## Prosjekt del 2 - Scala:

ActorSystem: Hierarchical group of actors that share common configuration options.

Responsible for creating new actors, locating actors within an actor system, and logging important events.

BankManager → `bankID { bankID : ActorRef }`  
`accounts { ActorRef }`  
↳ `create` —  
`find` —

ActorClass: a template that describes a state internal to the actor, and how the actor processes messages. Multiple actors can be created from the same actor class.

Account `accID`  
`bankID` → reference to Bank  
`balance`  
`transactions { }`

describes the behavior of an actor: how the actor responds to messages and communicates with other actors, encapsulates actor state, and defines the actor's startup and shutdown sequences.

Declaring as Actor when "Class — extends Actor"  
→ Has 1 method: receive which returns a `PartialFunction[Any, Unit]`: is used when an actor receives a message of Any type. If the PartialFunction is not defined for the message, the message is discarded.

Bank `bankID`  
`accounter` → to gen 4 units unique ID (Acc)  
Also has receive.

### Transactions:

