

MODEL-MAN-INTENT

3



```
interface View : MvpView {  
    fun initIntent(): Observable<ConversationId>  
    fun sendMessageIntent(): Observable<Message>  
    fun openProfileIntent(): Observable<ProfileId>  
    fun closeIntent(): Observable<Any>  
    fun render(viewState: ChatViewState)  
}
```

```
abstract class Presenter : MviBasePresenter<View, ChatViewState>()
```

verify(view).what???()

MODEL-VIEW-INTENT

```
interface View : MvpView {  
    fun initIntent(): Observable<ConversationId>  
    fun sendMessageIntent(): Observable<Message>  
    fun openProfileIntent(): Observable<ProfileId>  
    fun closeIntent(): Observable<Any>  
    fun render(viewState: ChatViewState)  
}
```

```
abstract class Presenter : MviBasePresenter<View, ChatViewState>()
```

```
    verify(view).what???
```

VIEW ROBOT

```
val renderedStates = mutableListOf<ChatViewState>()

val initSubject = PublishSubject.create<ConversationId>()
val sendMessageSubject = PublishSubject.create<Message>()
val openProfileSubject = PublishSubject.create<ProfileId>()

val view: ChatContract.View = object : ChatContract.View {
    override fun initIntent(): Observable<Long> = initSubject

    override fun sendMessageIntent(): Observable<Message> = sendMessageSubject

    override fun openProfileIntent(): Observable<ProfileId> = openProfileSubject

    override fun render(viewState: ChatViewState) {
        renderedStates.add(viewState)
    }
}

init {
    presenter.attachView(view)
}
```