

1. ScenarioManagement.cs 에서 Json 형태로 된 튜토리얼 Data를 읽어온다.
 - A. ScenarioManager.cs -> ReadScenarioData()
2. 인게임 진입시 필요한 튜토리얼 정보를 ScenarioGameManager.cs 로 전달한다.
3. ScenarioGameManager.cs 에서 Scripts의 Methods의 내용들을 순차적으로 실행시킵니다. Update문

[부연 설명]

jsonFile의 구조를 보면, 챕터 단위 배열로 이루어져 있습니다. HumanChapterDatas.json 참고

챕터는 튜토리얼 큰 한 흐름(세트)을 뜻합니다.

내부에 scripts라는 field가 있는데, 이 scripts를 순차적으로 실행하면서 튜토리얼이 동작하게 됩니다.

예를 들어, 아래 예시와 같은 scripts가 존재합니다.

Set_Tutorial, Stop_Invoke_NextTurn, Wait_Turn, Stop_orc_summon

이렇게 3개의 행동을 차례대로 수행하는 scripts입니다.

각각 ScenarioExecuteHandler를 상속받는 함수의 이름에 해당합니다.

내부적으로 ScenarioExecuteHandler 함수를 호출하고, args field의 값을 인자로 전달합니다.

```
{
  "name": "Set_Tutorial",
  "args" : ["on"]
},
{
  "name": "Stop_Invoke_NextTurn",
  "args": []
},
{
  "name": "Wait_Turn",
  "args": [
    "MULIGAN"
  ]
},
{
  "name": "Stop_orc_summon",
  "args": [
    "units",
    "all"
```

]

}