Samples

Certainly! Here are some **recent open-source card game projects** developed using **Kotlin and Jetpack Compose**:

1. **Card Game Animation**
   * **Description**: A sample card game app demonstrating complex animations using Jetpack Compose.
   * **Repository**: [MohamedRejeb/Card-Game-Animation](https://github.com/MohamedRejeb/Card-Game-Animation)
2. **Durak**
   * **Description**: An online card game implemented with Jetpack Compose and Material 3. Features include user authentication, Firestore integration, and more.
   * **Repository**: [zaurh/Durak](https://github.com/zaurh/Durak)
3. **Jetpack Compose Card Game Animation UI-Design**
   * **Description**: A tutorial video demonstrating card game animations in Jetpack Compose.
   * **Video**: [YouTube Tutorial](https://www.youtube.com/watch?v=Lz6xwXLXCA4)
4. **Simple Android Memory Game**
   * **Description**: A tutorial on building a memory card game with different themes using Jetpack Compose.
   * **Video**: [YouTube Tutorial](https://www.youtube.com/watch?v=B7--mnjOOI4)
5. **Final Fantasy Game**
   * **Video : <https://www.youtube.com/watch?v=VikeOza2X5E>**
6. **How to build simple asteroid game (Not a card game but can provide ideas)**
   * **Description :** [**https://dev.to/kotlin/how-i-built-an-asteroids-game-using-jetpack-compose-for-desktop-309l**](https://dev.to/kotlin/how-i-built-an-asteroids-game-using-jetpack-compose-for-desktop-309l)

git add .

git commit -m "Refactor: moved Card, Hand, and Combination to data/model"

git push origin refactor-architecture

com.example.nido/

│── data/

│ ├── model/

│ │ ├── Card.kt

│ │ ├── Combination.kt

│ │ ├── Hand.kt

│ ├── repository/

│ │ ├── CardRepository.kt

│── game/

│ ├── rules/

│ │ ├── GameRules.kt

│ │ ├── ScoringSystem.kt

│ ├── ai/

│ │ ├── PlayerAI.kt

│ │ ├── Heuristics.kt

│── ui/

│ ├── views/

│ │ ├── CardView.kt

│ │ ├── HandView.kt

│ │ ├── MatView.kt

│ │ ├── PlayerRow.kt

│ │ ├── DiscardPileView.kt

│ ├── screens/

│ │ ├── GameScreen.kt

│ │ ├── SettingsScreen.kt

│── viewmodel/

│ ├── GameViewModel.kt

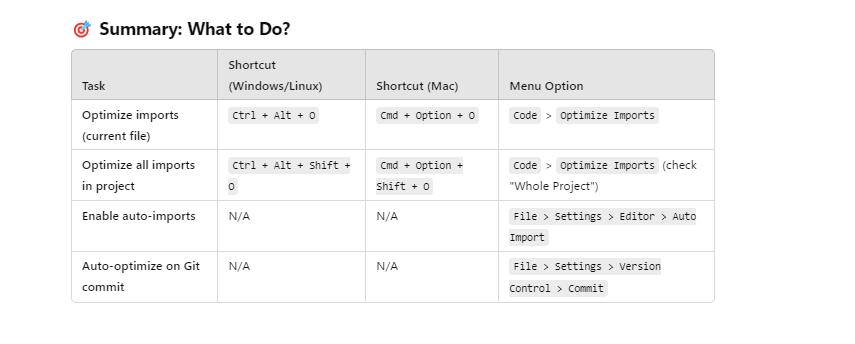
│── utils/

│ ├── SortingUtils.kt

│ ├── TestData.kt

│── MainActivity.kt

│── NidoApp.kt



CTRL + ALT + O : remove imports

CTRL+ALT+L : auto ident

0123ChatGPT4!

**🔹 Common Pictograms for Debugging**

Here are some commonly used pictograms for debugging:

**📌 General Debugging**

| **Emoji** | **Meaning** | **Unicode** |
| --- | --- | --- |
| 🔍 | Search/Inspect | U+1F50D |
| 🛠 | Fixing Issue | U+1F6E0 |
| ⚠️ | Warning | U+26A0 |
| ❌ | Error | U+274C |
| ✅ | Success | U+2705 |
| 🔄 | Reload/Updating | U+1F504 |
| ⏳ | Waiting | U+23F3 |

**🎮 Game State Debugging**

| **Emoji** | **Meaning** | **Unicode** |
| --- | --- | --- |
| 🎲 | Dice (Random/Turn) | U+1F3B2 |
| 🃏 | Card | U+1F0CF |
| 🎴 | Deck of Cards | U+1F3F4 |
| 🏆 | Winner | U+1F3C6 |
| 🗑 | Discard | U+1F5D1 |
| 🎮 | Game Controller | U+1F3AE |

**🔊 Audio Debugging**

| **Emoji** | **Meaning** | **Unicode** |
| --- | --- | --- |
| 🔊 | Sound On | U+1F50A |
| 🔇 | Sound Off | U+1F507 |

**👥 Players & Turn Management**

| **Emoji** | **Meaning** | **Unicode** |
| --- | --- | --- |
| 👤 | Single Player | U+1F464 |
| 👥 | Multiple Players | U+1F465 |
| ⏩ | Next Turn | U+23E9 |

**🔹 How to Use in println**

kotlin

CopierModifier

println("🔍 Inspecting GameState: $gameState")

println("🛠 Fixing player state: ${player.name}")

println("⚠️ Warning: Deck is almost empty! Cards left: ${deck.size}")

println("✅ Game successfully initialized!")

**🚀 Bonus: Format Logs for Readability**

kotlin

CopierModifier

println("🔥 [DEBUG] 🎮 GameState Update:")

println(" 📌 Screen: ${gameState.screen}")

println(" 🎮 Players: ${gameState.players}")

println(" 🎴 Deck: ${gameState.deck.size} cards")

println(" 🗑 Discard Pile: ${gameState.discardPile.size} cards")

println(" 🎲 Current Turn: Player ${gameState.currentPlayerIndex}")

**✅ Where Else Can You Use Them?**

* Logging with Log.d("TAG", "🔍 Debug Info: $data")
* Debugging println()
* UI (Text elements in Compose)
* Notifications & Toast messages

🚀 **Now you have a list of pictograms for deb**

🟨

🟡

⚠

❌

💀

❌