

## 1. 사전 준비 (Prerequisites)

봇을 실행하기 전에 디스코드 봇 토큰이 필요합니다.

### 가. 디스코드 봇 토큰 (Discord Bot Token)

1. **Discord Developer Portal** 로 이동하여 로그인합니다.
2. **New Application** 버튼을 클릭하고 봇의 이름을 정합니다 (예: 토너먼트 봇).
3. 생성된 애플리케이션의 **Bot** 탭으로 이동합니다.
4. **Add Bot** 버튼을 클릭하여 봇을 생성합니다.
5. **Reset Token** 버튼을 클릭하여 봇의 토큰을 복사합니다. 이 토큰은 매우 중요하므로 안전하게 보관해야 합니다.
6. **Privileged Gateway Intents** 섹션에서 INTENT 3개를 모두 활성화(enable)해야 합니다. 이는 봇이 멤버 목록을 보고 메시지 내용을 읽는 데 필요합니다.

---

## 2. 서버 배포 가이드 (Ubuntu VPS 기준)

이 가이드는 봇을 24시간 안정적으로 운영하기 위해 리눅스(Ubuntu) 가상 서버(VPS)에 배포하는 과정을 안내합니다.

### 2.1. 준비물

- 리눅스(Ubuntu) VPS: iwinv, AWS, GCP 등 클라우드 서비스에서 제공하는 가상 서버.
- 소스코드 압축 파일: 이 프로젝트 폴더 전체를 **tournament-bot.zip**과 같이 압축한 파일.
- SSH 접속 프로그램: Windows의 **PuTTY** 또는 macOS의 기본 터미널.

### 2.2. 서버 접속 및 기본 프로그램 설치

1. SSH로 서버 접속 터미널을 열고 아래 명령어로 VPS에 접속합니다.

```
ssh [사용자이름]@[서버_IP_주소]
```

2. 시스템 업데이트 패키지 목록을 최신 상태로 업데이트합니다.

```
sudo apt update && sudo apt upgrade -y
```

3. Node.js, npm, Unzip 설치 봇 실행에 필요한 Node.js 환경과 압축 해제 도구를 설치합니다.

```
sudo apt install -y nodejs npm unzip
```

4. PM2 설치 Node.js 애플리케이션(봇)을 24시간 실행하고 관리해주는 프로세스 매니저입니다.

```
sudo npm install -g pm2
```

## 2.3. 소스코드 업로드 및 설정

1. 소스코드 업로드 (SCP) 로컬 컴퓨터의 터미널에서 아래 명령어를 실행하여 소스코드 압축 파일을 서버로 전송합니다.

```
# scp [로컬 파일 경로] [서버 사용자이름]@[서버 IP주소]:[서버에 저장될 경로]
# 예시: 바탕화면의 tournament-bot.zip을 서버의 홈 디렉토리에 업로드
scp ~/Desktop/tournament-bot.zip ubuntu@123.45.67.89:/home/ubuntu/
```

2. 서버에서 압축 풀기 다시 서버에 접속한 SSH 터미널로 돌아와서 압축을 풉니다.

```
# 파일명이 다르다면 실제 파일명으로 변경하세요.
unzip tournament-bot.zip
# 생성된 프로젝트 폴더로 이동합니다.
cd 250828-tournament-bot
```

3. 의존성 설치 및 환경 설정

- 봇 실행에 필요한 라이브러리를 설치합니다.

```
npm install
```

- **.env** 파일을 서버 환경에 맞게 생성하고 내용을 채웁니다.

```
nano .env
```

4. 데이터베이스 설정 Prisma를 사용하여 SQLite 데이터베이스를 초기화합니다.

```
npm run db:generate
npm run db:push
```

## 2.4. PM2로 봇 실행하기

1. PM2로 봇 시작

```
npm run deploy
```

2. 실행 상태 확인

```
pm2 list
```

---

`tournament-bot` 프로세스가 `online` 상태로 표시되면 성공입니다.

3. 서버 재부팅 시 자동 실행 설정 아래 명령어를 실행해두면, 서버가 재부팅되어도 PM2가 봇을 자동으로 다시 실행시켜 줍니다.

```
pm2 startup
# 화면에 나오는 `sudo env ...`로 시작하는 명령어를 복사하여 한 번 더 실행해주세요.
pm2 save
```

## 2.5. 봇 관리 명령어 (PM2)

- 로그 확인: `pm2 logs tournament-bot`
- 봇 재시작: `pm2 restart tournament-bot`
- 봇 중지: `pm2 stop tournament-bot`
- 프로세스 목록에서 제거: `pm2 delete tournament-bot`

---

## 3. 로컬 개발 환경 설정

### 3.1. `.env` 파일 설정

프로젝트의 최상위 폴더에 `.env`라는 이름의 파일을 생성하고 아래 내용을 붙여넣으세요. 그리고 `사전 준비` 단계에서 얻은 값들을 입력합니다.

```
# 디스코드 봇 토큰
DISCORD_TOKEN=여기에_디스코드_봇_토큰을_붙여넣으세요

# 데이터베이스 URL (SQLite 사용)
DATABASE_URL="file:./dev.db"
```

### 3.2. 로컬 실행

```
# 의존성 설치
npm install

# Prisma 데이터베이스 설정
npm run db:generate
npm run db:push

# 봇 실행
npm start
```