🎬 Telegram Media Downloader Bot

A powerful Telegram bot that downloads videos and audio from **YouTube, Newgrounds, Instagram, TikTok, Twitter/X, Facebook, Reddit,** and **1500+ other platforms** using yt-dlp.

🌟 Features

Core Functionality
- ✅ **1500+ Supported Sites** - YouTube, Newgrounds, Instagram, TikTok, Twitter, Facebook, Reddit, Vimeo, and more
- 🎥 **Multiple Quality Options** - Best, 1080p, 720p, 480p
- 🎵 **Audio Extraction** - Download audio-only in MP3 format (320kbps)
- 📱 **Mobile Friendly** - Works seamlessly on all Telegram clients
- 👥 **Group Chat Support** - Use in private chats or groups
- 🔐 **User Whitelist** - Optional user authorization
- 📊 **Progress Tracking** - Real-time download progress
- 🧹 **Auto Cleanup** - Automatic file deletion after upload

Platform-Specific Features
- **YouTube**: Videos, Shorts, Playlists, Live streams
- **Newgrounds**: Animations, Videos, Game trailers
- **Instagram**: Posts, Reels, Stories, IGTV, Carousels
- **TikTok**: Videos with or without watermark
- **Twitter/X**: Videos and GIFs
- **Reddit**: Video posts with audio
- **Facebook**: Public videos
- **And 1500+ more!**

📋 Quick Start

Prerequisites
- Python 3.8 or higher
- FFmpeg (for audio/video processing)
- Telegram Bot Token from [@BotFather](https://t.me/BotFather)

Installation

1. **Clone or download the repository**
   ```bash
   git clone <repository-url>
   cd telegram-media-bot
   ```

2. **Install dependencies**
   ```bash
   pip install -r requirements.txt
   ```

3. **Install FFmpeg**
   - **Windows**: Download from [ffmpeg.org](https://ffmpeg.org/download.html) and add to PATH
   - **macOS**: `brew install ffmpeg`

- **Linux**: `sudo apt install ffmpeg`

4. **Configure your bot**
   ```bash
   cp .env.example .env
   # Edit .env and add your BOT_TOKEN
   ```

5. **Run the bot**
   ```bash
   python telegram_media_bot.py
   ```

🚀 Usage

Basic Commands

Command	Description
`/start`	Show welcome message and features
`/help`	Display detailed help and instructions
`/quality`	Select video quality (Best/1080p/720p/480p)
`/audio`	Switch to audio-only mode
`/video`	Switch to video mode (default)

How to Download

1. Send any video URL to the bot
2. The bot will automatically detect and process it
3. Choose quality if prompted
4. Wait for download and upload
5. Receive your media file!

Example URLs

```

# YouTube
https://www.youtube.com/watch?v=dQw4w9WgXcQ

# Newgrounds
https://www.newgrounds.com/portal/view/854273

# Instagram
https://www.instagram.com/p/CXpxSyOrWCA/

# TikTok
https://www.tiktok.com/@user/video/1234567890

# Twitter
https://twitter.com/user/status/1234567890
```

🔧 Configuration

Environment Variables

Create a `.env` file or set environment variables:

```bash
# Required
BOT_TOKEN=your_bot_token_here

# Optional
DOWNLOAD_PATH=./downloads        # Where to store temporary files
MAX_FILE_SIZE_MB=50              # Telegram upload limit (50MB non-premium, 2GB premium)
ALLOWED_USERS=                   # Comma-separated user IDs (empty = all users allowed)
```

Get Your Bot Token

1. Open Telegram and search for [@BotFather](https://t.me/BotFather)
2. Send `/newbot` command
3. Follow the prompts to create your bot
4. Copy the token (format: `123456789:ABCdefGHIjklMNOpqrsTUVwxyz`)

Optional: Restrict Access

To limit bot access to specific users:

1. Get your Telegram user ID from [@userinfobot](https://t.me/userinfobot)
2. Add to `.env`:
   ```
   ALLOWED_USERS=123456789,987654321
   ```

🐳 Docker Deployment

Using Docker Compose (Recommended)

1. **Create `.env` file**
   ```bash
   BOT_TOKEN=your_token_here
   MAX_FILE_SIZE_MB=50
   ```

2. **Start the bot**
   ```bash
   docker-compose up -d
   ```

3. **View logs**
   ```bash

```
   docker-compose logs -f
   ```

4. **Stop the bot**
   ```bash
   docker-compose down
   ```

Using Docker directly

```bash
docker build -t telegram-media-bot .
docker run -d \
  --name media-bot \
  -e BOT_TOKEN=your_token_here \
  -e MAX_FILE_SIZE_MB=50 \
  -v $(pwd)/downloads:/downloads \
  telegram-media-bot
```

📚 Supported Platforms

The bot uses **yt-dlp** which supports **1500+ websites**:

Video Platforms
- YouTube (videos, shorts, playlists, live streams)
- Vimeo
- Dailymotion
- Twitch
- Newgrounds ⭐

Social Media
- Instagram (posts, reels, stories, IGTV)
- TikTok
- Twitter/X
- Facebook
- Reddit
- Pinterest
- LinkedIn
- Snapchat

More Platforms
- Bilibili
- Soundcloud
- Mixcloud
- Bandcamp
- Archive.org
- And many more!

Full list: [yt-dlp supported sites](https://github.com/yt-dlp/yt-dlp/blob/master/supportedsites.md)

⚙️ Advanced Features

Custom Quality Selection

Users can choose from multiple quality options:
- **Best**: Highest available quality
- **1080p**: Full HD
- **720p**: HD
- **480p**: SD (smaller file size)

Audio-Only Mode

Extract audio in high-quality MP3 (320kbps):
1. Send `/audio` to enable audio mode
2. Send any video URL
3. Receive MP3 file

Playlist Support

Send any playlist URL to download:
- YouTube playlists
- Soundcloud sets
- Spotify playlists (if supported)

Note: Large playlists may take time and hit file size limits.

Progress Updates

The bot provides real-time status updates:
- ⏳ Processing
- 📥 Downloading
- 📤 Uploading
- ✅ Complete

🛠️ Troubleshooting

Common Issues

Bot doesn't respond
- Verify bot token is correct
- Check bot is running: `python telegram_media_bot.py`
- Ensure `/start` was sent first

"HTTP Error 403" or download fails
- Video may be private or geo-restricted
- Try updating yt-dlp: `pip install -U yt-dlp`
- Some sites require authentication

"File too large"
- Select lower quality: `/quality` → 720p or 480p
- Use audio-only mode: `/audio`

- Telegram limits: 50MB (regular), 2GB (premium)

FFmpeg not found
- Install FFmpeg: `sudo apt install ffmpeg` (Linux) or `brew install ffmpeg` (macOS)
- Verify installation: `ffmpeg -version`
- Add to PATH on Windows

Newgrounds videos fail
- Update yt-dlp to latest version
- Some content requires Newgrounds account
- Check if video is publicly accessible

Debug Mode

Enable detailed logging:
```python
logging.basicConfig(level=logging.DEBUG)
```

Getting Help

1. Check the SETUP_GUIDE.md
2. Review [yt-dlp documentation](https://github.com/yt-dlp/yt-dlp)
3. Search existing GitHub issues
4. Create a new issue with error logs

📝 Legal & Ethics

Important Notice

⚠️ **This bot is for educational and personal use only.**

- Respect copyright laws and platform terms of service
- Only download content you have rights to access
- Some platforms prohibit automated downloads
- Be aware of regional content restrictions
- Use responsibly and ethically

Recommended Use Cases

✅ **Appropriate Uses:**
- Creative Commons licensed content
- Personal backups of your own content
- Educational and research purposes
- Content with explicit download permissions
- Archive preservation

❌ **Not Recommended:**
- Copyrighted commercial content
- Content for redistribution
- Bypassing paywalls

- Violating platform ToS

🤝 Contributing

Contributions are welcome! Here's how you can help:

1. **Report Bugs**: Open an issue with details
2. **Suggest Features**: Share your ideas
3. **Submit PRs**: Improve code or documentation
4. **Test**: Try on different platforms
5. **Share**: Tell others about the project

Development

```bash
# Clone repo
git clone <repository-url>
cd telegram-media-bot

# Create virtual environment
python -m venv venv
source venv/bin/activate  # Linux/Mac
venv\Scripts\activate     # Windows

# Install dependencies
pip install -r requirements.txt

# Run in development
python telegram_media_bot.py
```

📊 Project Structure

```
telegram-media-bot/
├── telegram_media_bot.py   # Main bot code
├── requirements.txt        # Python dependencies
├── .env.example            # Environment variables template
├── README.md               # This file
├── SETUP_GUIDE.md          # Detailed setup instructions
├── Dockerfile              # Docker configuration
├── docker-compose.yml      # Docker Compose setup
└── downloads/              # Temporary download directory
```

🔄 Updates

Keep your bot up to date:

```bash
# Update yt-dlp (recommended monthly)
```

```
pip install -U yt-dlp

# Update python-telegram-bot
pip install -U python-telegram-bot

# Update all dependencies
pip install -U -r requirements.txt
```

📈 Performance Tips

1. **VPS Hosting**: Deploy on a VPS for 24/7 availability
2. **SSD Storage**: Faster downloads and uploads
3. **Good Internet**: Required for large files
4. **Monitoring**: Set up logging and alerts
5. **Rate Limiting**: Prevent abuse if bot is public

🌐 Deployment Options

Cloud Platforms
- **Heroku**: Free tier available (with limitations)
- **DigitalOcean**: $5/month VPS
- **AWS EC2**: Free tier for 12 months
- **Google Cloud**: $300 free credit
- **Railway**: Simple deployment

Self-Hosted
- **Raspberry Pi**: Low-power 24/7 operation
- **Home Server**: Full control
- **NAS**: Many support Docker

📜 License

This project is provided as-is for educational purposes. Use responsibly and in compliance with all applicable laws and terms of service.

🙏 Acknowledgments

Built with:
- [yt-dlp](https://github.com/yt-dlp/yt-dlp) - Media download engine
- [python-telegram-bot](https://github.com/python-telegram-bot/python-telegram-bot) - Telegram Bot framework
- [FFmpeg](https://ffmpeg.org/) - Audio/video processing

Inspired by the @tMediaDownloaderBot and similar Telegram media downloaders.

📞 Support

- **Documentation**: Check SETUP_GUIDE.md
- **Issues**: GitHub Issues page
- **Updates**: Follow repository for updates
- **Community**: Join discussions

⭐ Star This Project

If you find this useful, please star the repository to show your support!

Made with ❤️ for the Telegram community

Version: 1.0.0 | **Last Updated**: November 2024