class Posts extends StatefulWidget

const Posts({Key? key, required this.data}) : super(key: key);

final data;

class _Posts extends State<Posts>

class CardPosts

class SubredditPage

---

class MyApp extends StatelessWidget

const MyApp({Key? key}) : super(key: key);

class MainRouter extends StatefulWidget

const MainRouter({Key? key}) : super(key: key);

class _MainRouter extends State<MainRouter>

int selectedIndex;
Widget _login;
Widget _homePage;

void switchRouter(int value);

---

class SubredditPage extends StatefulWidget

const SubredditPage({Key? key, required this.idSub}) : super(key: key);

final idSub;

class _SubredditPage extends State<SubredditPage>

bool selected;
bool isLoading;
bool isSubLoading;
bool isLoadingPosts;
List<HotPosts> hotPosts;
String dropdownValue;
String after;
late Subreddit data;
List<Widget> widFirstIno;

void initState();
Future loadDataPosts(limit) async;
Future loadData() async;
Future onSubPressed() async;

class Posts

class CardPosts

HotPosts

final String title;
final String subreddit;
final String selfText;
final void selfTextHtml;
final String subredditNamePrefixed;
final String urlSubredditToPost;
final int numComments;
final int score;
final String permalink;
final String id;
final String author;
final bool isSaved;
final int likes;
late final String urlAvatarSubreddit;
final List<dynamic> listUrlImage;
final List<dynamic> listUrlGif;
final List<dynamic> listUrlVideo;
final List<dynamic> listUrlGallery;

HotPosts({...});
factory HotPosts.fromJson(Map<String, dynamic> json);

---

class LoginPage extends StatefulWidget

const LoginPage({Key? key, required this.switchMainRouter}) : super(key: key);

class _LoginPage extends State<LoginPage>

static const String clientId;
static const String clientSecret;
static const String scope;
final LocalStorage storage;
bool isOnApi;

void initState();
Future<void> requestToken(String url) async;

---

class MyHomePage extends StatefulWidget

const MyHomePage({Key? key, required this.switchMainRouter}) : super(key: key);

class MyHomePageState extends State<MyHomePage>

int selectedIndex;
final Widget _home;
final Widget _myProfile;
final Widget _search;
static const timeout;
static const ms;

Timer startTimeout([int? milliseconds]);
void handleTimeout();
String getNameApp();
Widget getBody();
void onTapHandler(int index);

Subreddit

final String urlIconImg;
final String name;
final String title;
final int subscribers;
final int activeAccount;
final String description;
final String urlBanner;
final bool isSubscriber;

Subreddit({});
factory Subreddit.fromJson(Map<String, dynamic> json);

---

class CardPosts extends StatefulWidget

const CardPosts(
  {Key? key,
  required this.data,
  required this.isDisableComment,
  required this.isDisableSubAction})
  : super(key: key);

final data;
final bool isDisableComment;
final bool isDisableSubAction;

class _CardPosts extends State<CardPosts>

bool isSaved;
int valueVote;
int totalVote;

void initState();
Widget _getGallery();
List<Widget> _getImage(elem);
Widget _getSelfText();
Future saveOrNot() async;
Future votePosts(int value) async;

---

class Home extends StatefulWidget

const Home({Key? key}) : super(key: key);

class _Home extends State<Home>

List<HotPosts> hotPosts;
String dropdownValue;
String after;
bool isLoading;

void initState();
Future loadDataPosts(limit) async;

class Posts

class SubredditPage

class CardPosts

---

class MyProfile extends StatefulWidget

const MyProfile({Key? key}) : super(key: key);

class _MyProfile extends State<MyProfile>

late Future<User> futureUser;
bool selected = true;

void initState()
Widget buildImage(imagePath)
Widget buildCircle();
Widget buildName(name);
Widget buildKarmaCoin(karma, coins, friends);

class SettingsProfile extends StatefulWidget

const SettingsProfile({Key? key}) : super(key: key);

class _SettingsProfile extends State<SettingsProfile>

bool isLoading;
bool acceptFollowers;
bool showPresence;
bool isSwitched_5;
bool autoplay;
bool adultContent;
bool beta;
bool contentVisibility;
bool emailRequest;
bool emailComments;
bool emailUpvotePost;
bool emailPrivateMessage;
bool emailUserMention;

void initState();
Future loadSettings() async;
Future patchSettings() async;

---

class Search extends StatefulWidget

const Search({Key? key}) : super(key: key);

class _Search extends State<Search>

bool isLoading;
String searchValue;
List<DataSearch> sub;

void initState();
Future loadSearch() async;

class SubredditPage

User

final String displayName;
final String avatarUrl;
final String idUser;
final String bannerImg;
final String description;
final int coins;
final int karma;
final int numberFriend;
final bool autoplay;
final bool acceptFollowers;
final bool showMedia;
final bool showPresence;
final bool nsfw;

User({});
factory User.fromJson(Map<String, dynamic> json);

DataSearch

final String name;
final String imgIcon;

DataSearch({});
factory DataSearch.fromJson(Map<String, dynamic> json);