

PagerAdapter Inheritance

PagerAdapter

```
abstract int      getCount();
    Object  instantiateItem(ViewGroup container, int position) {thr Ex}
    void    destroyItem(ViewGroup container, int position, Object object) {thr Ex}
abstract boolean  isViewFromObject(View view, Object object);
    int      getItemPosition(Object object) {return POSITION_UNCHANGED;}
    void     notifyDataSetChanged() {...}
```

MyPlainPagerAdapter

```
int      getCount() {return characterAdapter.getCount();}
Object  instantiateItem(ViewGroup, int) {...}
void    destroyItem(ViewGroup, int, Object) {...}
boolean  isViewFromObject(View, Object) {...}
```

FragmentPagerAdapter

```
    Object  instantiateItem(ViewGroup, int) {getItem(int)}
abstract Fragment getItem(int position);
    void    destroyItem(ViewGroup, int, Object) {...}
    boolean  isViewFromObject(View, Object) {...}
    long     getItemId(int) {...}
```

MyFragPagerAdapter (Straight)

```
int      getCount() {return characterAdapter.getCount();}
Fragment getItem(int position) {return CharacterFragment.newInstance(characterAdapter.getCharacterAt(position));}
```

MyFragPagerAdapter (Kinked)

```
int      getCount() {same}
Fragment getItem(int position) {...}
int      getItemPosition(Object object) {return characterAdapter.getItemPosition(character);}
long     getItemId(int position) {...}
void     changeChildTo(String parent, String nextChild) {characterAdapter.changeChildTo(parent, nextChild);}
```

FragmentStatePagerAdapter

```
    Object  instantiateItem(ViewGroup, int) {getItem(int)}
abstract Fragment getItem(int position);
    void    destroyItem(ViewGroup, int, Object) {...}
    boolean  isViewFromObject(View, Object) {...}
```

MyFragStatePagerAdapter (Kinked)

```
int      getCount() {same as MyFragPagerAdapter}
Fragment getItem(int position) {same as MyFragPagerAdapter}
int      getItemPosition(Object object) {same as MyFragPagerAdapter}
void     changeChildTo(String parent, String nextChild) {same as MyFragPagerAdapter}
```

key: thr Ex = throw Exception. All methods are public. '...' represents code