digiMED Documentation

Release

Mind Optimizers

Nov 26, 2020

CONTENTS:

CHAPTER

ONE

SOURCE

1.1 Account package

1.1.1 Subpackages

1.1.2 Submodules

1.1.3 Account.admin module

1.1.4 Account.apps module

class Account.apps.**AccountConfig**(*app_name*, *app_module*)

Bases: django.apps.config.AppConfig

name = 'Account'

1.1.5 Account.models module

class Account.models.**UserProfile**(*id*, *user*, *profile_type*, *age*)

Bases: django.db.models.base.Model

exception DoesNotExist

Bases: django.core.exceptions.ObjectDoesNotExist

exception MultipleObjectsReturned

Bases: django.core.exceptions.MultipleObjectsReturned

age

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

doctor_set

Accessor to the related objects manager on the reverse side of a many-to-one relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

Parent.children is a ReverseManyToOneDescriptor instance.

Most of the implementation is delegated to a dynamically defined manager class built by create_forward_many_to_many_manager() defined below.

id
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

objects = <django.db.models.manager.Manager object>

patient_set
Accessor to the related objects manager on the reverse side of a many-to-one relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

`Parent.children` is a `ReverseManyToOneDescriptor` instance.

Most of the implementation is delegated to a dynamically defined manager class built by `create_forward_many_to_many_manager()` defined below.

profile_type
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

user
Accessor to the related object on the forward side of a one-to-one relation.

In the example:

```
class Restaurant(Model):
    place = OneToOneField(Place, related_name='restaurant')
```

`Restaurant.place` is a `ForwardOneToOneDescriptor` instance.

user_id

1.1.6 Account.tests module

1.1.7 Account.urls module

1.1.8 Account.views module

Account.views.**login**(*request*)
Logs in the users to their respective accounts

Parameters **request** – Default HttpRequest object

Returns Rendered HTML Webpage

Account.views.**logout**(*request*)
Redirects the user to the register page when they logout of the respective account

Parameters **request** – Default HttpRequest object

Returns Rendered HTML Webpage

Account.views.**register**(*request*)
Takes the information of the users (doctor/patients) and fills the same in the database

Parameters **request** – Default HttpRequest object

Returns Rendered HTML Webpage

1.1.9 Module contents

1.2 db_models module

1.3 digiMED package

1.3.1 Submodules

1.3.2 digiMED.asgi module

ASGI config for digiMED project.

It exposes the ASGI callable as a module-level variable named `application`.

For more information on this file, see https://docs.djangoproject.com/en/3.1/howto/deployment/asgi/

1.3.3 digiMED.settings module

Django settings for digiMED project.

Generated by 'django-admin startproject' using Django 3.1.2.

For more information on this file, see https://docs.djangoproject.com/en/3.1/topics/settings/

For the full list of settings and their values, see https://docs.djangoproject.com/en/3.1/ref/settings/

1.3.4 digiMED.urls module

digiMED URL Configuration

The *urlpatterns* list routes URLs to views. For more information please see: https://docs.djangoproject.com/en/3.1/topics/http/urls/

Examples: Function views

1. Add an import: from my_app import views
2. Add a URL to urlpatterns: path('', views.home, name='home')

Class-based views

1. Add an import: from other_app.views import Home
2. Add a URL to urlpatterns: path('', Home.as_view(), name='home')

Including another URLconf

1. Import the include() function: from django.urls import include, path
2. Add a URL to urlpatterns: path('blog/', include('blog.urls'))

1.3.5 digiMED.wsgi module

WSGI config for digiMED project.

It exposes the WSGI callable as a module-level variable named `application`.

For more information on this file, see https://docs.djangoproject.com/en/3.1/howto/deployment/wsgi/

1.3.6 Module contents

1.4 doctor package

1.4.1 Subpackages

1.4.2 Submodules

1.4.3 doctor.admin module

1.4.4 doctor.apps module

class doctor.apps.**DoctorConfig**(*app_name*, *app_module*)
Bases: django.apps.config.AppConfig

name = 'doctor'

1.4.5 doctor.models module

class doctor.models.**doctorRequestsTable**(*id*, *pid*, *docid*)
Bases: django.db.models.base.Model

exception DoesNotExist
Bases: django.core.exceptions.ObjectDoesNotExist

exception MultipleObjectsReturned
Bases: django.core.exceptions.MultipleObjectsReturned

docid
Accessor to the related object on the forward side of a many-to-one or one-to-one (via ForwardOneToOneDescriptor subclass) relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

Child.parent is a ForwardManyToOneDescriptor instance.

docid_id

id
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

objects = <django.db.models.manager.Manager object>

pid
Accessor to the related object on the forward side of a many-to-one or one-to-one (via ForwardOneToOneDescriptor subclass) relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

Child.parent is a ForwardManyToOneDescriptor instance.

pid_id

1.4.6 doctor.tests module

1.4.7 doctor.urls module

1.4.8 doctor.views module

doctor.views.**doctorRequest**(*request*)
It will store the patient id whose health records the current user/doctor wants to access

Parameters **request** – Default HttpRequest object

Returns Rendered HTML Webpage

doctor.views.**doctorSignup**(*request*)
It registers the details of the doctor and redirects the user to home page

Parameters **request** – Default HttpRequest object

Returns Rendered HTML Webpage

doctor.views.**doctorshowhisttory**(*request*, *patientid*)
It will show the health record of the patient

Parameters **request** – Default HttpRequest object

Returns Rendered HTML Webpage

doctor.views.**index**(*request*)
It will fetch the patient id whose health records are to be shown

Parameters **request** – Default HttpRequest object

Returns Rendered HTML Webpage

doctor.views.**login**(*request*)
Backup login page for doctor

Parameters **request** – Default HttpRequest object

Returns Rendered HTML Webpage

doctor.views.**logout**(*request*)
Logs out the user from his/her account

Parameters **request** – Default HttpRequest object

Returns Rendered HTML Webpage

doctor.views.**saveprescription**(*request*, *patient*)
It will create a prescription record linking the doctor with his/her patient for whom the medicines are prescribed

Parameters **request** – Default HttpRequest object

Returns Rendered HTML Webpage

doctor.views.**saveprescriptionmedicine**(*request*)
It will save the medicines prescribed by the doctor to his/her patient in the database

Parameters **request** – Default HttpRequest object

Returns Rendered HTML Webpage

1.4.9 Module contents

1.5 lab package

1.5.1 Subpackages

1.5.2 Submodules

1.5.3 lab.admin module

1.5.4 lab.apps module

class lab.apps.**LabConfig**(*app_name*, *app_module*)
Bases: django.apps.config.AppConfig

name = 'lab'

1.5.5 lab.models module

class lab.models.**LabTest**(*id*, *pid*, *date*, *test*)
Bases: django.db.models.base.Model

exception DoesNotExist
Bases: django.core.exceptions.ObjectDoesNotExist

exception MultipleObjectsReturned
Bases: django.core.exceptions.MultipleObjectsReturned

date
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

id
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

objects = <django.db.models.manager.Manager object>

pid
Accessor to the related object on the forward side of a many-to-one or one-to-one (via ForwardOneToOneDescriptor subclass) relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

Child.parent is a ForwardManyToOneDescriptor instance.

pid_id

test
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

1.5.6 lab.tests module

1.5.7 lab.urls module

1.5.8 lab.views module

lab.views.**index**(*request*)
It fetches the pending requests from the database and shows on the main page of lab technician and also inserts into the database any of the lab test details filled by the lab technician

Parameters **request** – Default HttpRequest object

Returns Rendered HTML Webpage

lab.views.**log**(*request*)
Backup login page for lab technician

Parameters **request** – Default HttpRequest object

Returns Rendered HTML Webpage

lab.views.**test**(*request*)
It will show the webpage containing the form to be filled for the respective lab test

Parameters **request** – Default HttpRequest object

Returns Rendered HTML Webpage

1.5.9 Module contents

1.6 manage module

Django's command-line utility for administrative tasks.

manage.**main**()
Run administrative tasks.

1.7 patient package

1.7.1 Subpackages

1.7.2 Submodules

1.7.3 patient.admin module

1.7.4 patient.apps module

class patient.apps.**PatientConfig**(*app_name*, *app_module*)
Bases: django.apps.config.AppConfig

name = 'patient'

1.7.5 patient.models module

class patient.models.**Basicmetabolismpanel**(*id*, *pid*, *date*, *glucose*, *creatinine*, *sodium*, *potassium*, *chloride*, *carbon_dioxide*)

Bases: django.db.models.base.Model

exception DoesNotExist

Bases: django.core.exceptions.ObjectDoesNotExist

exception MultipleObjectsReturned

Bases: django.core.exceptions.MultipleObjectsReturned

carbon_dioxide

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

chloride

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

creatinine

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

date

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

get_next_by_date(*, *field=<django.db.models.fields.DateField: date>*, *is_next=True*, ***kwargs*)

get_previous_by_date(*, *field=<django.db.models.fields.DateField: date>*, *is_next=False*, ***kwargs*)

glucose

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

id

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

objects = <django.db.models.manager.Manager object>

pid

Accessor to the related object on the forward side of a many-to-one or one-to-one (via ForwardOneToOneDescriptor subclass) relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

Child.parent is a ForwardManyToOneDescriptor instance.

pid_id

potassium

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

sodium

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

class patient.models.**Commonbloodcount**(*id*, *pid*, *date*, *wbc*, *rbc*, *hemoglobin*, *hematocrit*, *mcv*, *mch*, *mchc*, *rdw*, *platelets*, *neurophils*, *lymphs*, *monocytes*)

Bases: django.db.models.base.Model

exception DoesNotExist

Bases: django.core.exceptions.ObjectDoesNotExist

exception MultipleObjectsReturned

Bases: django.core.exceptions.MultipleObjectsReturned

date

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

get_next_by_date(*, *field=<django.db.models.fields.DateField: date>*, *is_next=True*, ***kwargs*)

get_previous_by_date(*, *field=<django.db.models.fields.DateField: date>*, *is_next=False*, ***kwargs*)

hematocrit

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

hemoglobin

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

id

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

lymphs

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

mch

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

mchc

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

mcv

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

monocytes

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

neurophils

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

objects = <django.db.models.manager.Manager object>

pid

Accessor to the related object on the forward side of a many-to-one or one-to-one (via ForwardOneToOneDescriptor subclass) relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

`Child.parent` is a `ForwardManyToOneDescriptor` instance.

pid_id

platelets
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

rbc
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

rdw
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

wbc
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

class `patient.models.`**Doctor**(*did*, *name*, *surname*, *gender*, *dob*, *country*, *profile_id*)
Bases: `django.db.models.base.Model`

exception DoesNotExist
Bases: `django.core.exceptions.ObjectDoesNotExist`

exception MultipleObjectsReturned
Bases: `django.core.exceptions.MultipleObjectsReturned`

country
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

did
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

dob
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

doctorrequeststable_set
Accessor to the related objects manager on the reverse side of a many-to-one relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

`Parent.children` is a `ReverseManyToOneDescriptor` instance.

Most of the implementation is delegated to a dynamically defined manager class built by `create_forward_many_to_many_manager()` defined below.

gender
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

name
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

objects = <django.db.models.manager.Manager object>

patientgranttable_set
Accessor to the related objects manager on the reverse side of a many-to-one relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

`Parent.children` is a `ReverseManyToOneDescriptor` instance.

Most of the implementation is delegated to a dynamically defined manager class built by `create_forward_many_to_many_manager()` defined below.

prescription_set
Accessor to the related objects manager on the reverse side of a many-to-one relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

`Parent.children` is a `ReverseManyToOneDescriptor` instance.

Most of the implementation is delegated to a dynamically defined manager class built by `create_forward_many_to_many_manager()` defined below.

profile_id
Accessor to the related object on the forward side of a many-to-one or one-to-one (via ForwardOneToOneDescriptor subclass) relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

`Child.parent` is a `ForwardManyToOneDescriptor` instance.

profile_id_id

surname
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

class patient.models.**Medication**(*medid*, *name*, *generic_name*)
Bases: `django.db.models.base.Model`

exception DoesNotExist
Bases: `django.core.exceptions.ObjectDoesNotExist`

exception MultipleObjectsReturned
Bases: `django.core.exceptions.MultipleObjectsReturned`

generic_name
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

medid
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

name
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

objects = <django.db.models.manager.Manager object>

prescriptmed_set
Accessor to the related objects manager on the reverse side of a many-to-one relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

`Parent.children` is a `ReverseManyToOneDescriptor` instance.

Most of the implementation is delegated to a dynamically defined manager class built by `create_forward_many_to_many_manager()` defined below.

class patient.models.**Mri**(*id*, *pid*, *date*, *mri*)
Bases: django.db.models.base.Model

exception DoesNotExist
Bases: django.core.exceptions.ObjectDoesNotExist

exception MultipleObjectsReturned
Bases: django.core.exceptions.MultipleObjectsReturned

date
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

get_next_by_date(*, *field=<django.db.models.fields.DateField: date>*, *is_next=True*, ***kwargs*)

get_previous_by_date(*, *field=<django.db.models.fields.DateField: date>*, *is_next=False*, ***kwargs*)

id
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

mri
Just like the FileDescriptor, but for ImageFields. The only difference is assigning the width/height to the width_field/height_field, if appropriate.

objects = <django.db.models.manager.Manager object>

pid
Accessor to the related object on the forward side of a many-to-one or one-to-one (via ForwardOneToOneDescriptor subclass) relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

`Child.parent` is a `ForwardManyToOneDescriptor` instance.

pid_id

class patient.models.**Patient**(*pid*, *name*, *surname*, *gender*, *date*, *country*, *street_address*, *city*, *state*, *postal*, *profile_id*)

Bases: `django.db.models.base.Model`

exception DoesNotExist

Bases: `django.core.exceptions.ObjectDoesNotExist`

exception MultipleObjectsReturned

Bases: `django.core.exceptions.MultipleObjectsReturned`

basicmetabolismpanel_set

Accessor to the related objects manager on the reverse side of a many-to-one relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

`Parent.children` is a `ReverseManyToOneDescriptor` instance.

Most of the implementation is delegated to a dynamically defined manager class built by `create_forward_many_to_many_manager()` defined below.

city

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

commonbloodcount_set

Accessor to the related objects manager on the reverse side of a many-to-one relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

`Parent.children` is a `ReverseManyToOneDescriptor` instance.

Most of the implementation is delegated to a dynamically defined manager class built by `create_forward_many_to_many_manager()` defined below.

country

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

date

A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

doctorrequeststable_set

Accessor to the related objects manager on the reverse side of a many-to-one relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

`Parent.children` is a `ReverseManyToOneDescriptor` instance.

Most of the implementation is delegated to a dynamically defined manager class built by `create_forward_many_to_many_manager()` defined below.

gender
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

get_next_by_date(*, *field=<django.db.models.fields.DateField: date>*, *is_next=True*, ***kwargs*)

get_previous_by_date(*, *field=<django.db.models.fields.DateField: date>*, *is_next=False*, ***kwargs*)

labtest_set
Accessor to the related objects manager on the reverse side of a many-to-one relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

`Parent.children` is a `ReverseManyToOneDescriptor` instance.

Most of the implementation is delegated to a dynamically defined manager class built by `create_forward_many_to_many_manager()` defined below.

mri_set
Accessor to the related objects manager on the reverse side of a many-to-one relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

`Parent.children` is a `ReverseManyToOneDescriptor` instance.

Most of the implementation is delegated to a dynamically defined manager class built by `create_forward_many_to_many_manager()` defined below.

name
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

objects = <django.db.models.manager.Manager object>

patientgranttable_set
Accessor to the related objects manager on the reverse side of a many-to-one relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

`Parent.children` is a `ReverseManyToOneDescriptor` instance.

Most of the implementation is delegated to a dynamically defined manager class built by `create_forward_many_to_many_manager()` defined below.

pid
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

postal
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

prescription_set
Accessor to the related objects manager on the reverse side of a many-to-one relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

`Parent.children` is a `ReverseManyToOneDescriptor` instance.

Most of the implementation is delegated to a dynamically defined manager class built by `create_forward_many_to_many_manager()` defined below.

profile_id
Accessor to the related object on the forward side of a many-to-one or one-to-one (via ForwardOneToOneDescriptor subclass) relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

`Child.parent` is a `ForwardManyToOneDescriptor` instance.

profile_id_id

state
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

street_address
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

surname
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

xray_set
Accessor to the related objects manager on the reverse side of a many-to-one relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

`Parent.children` is a `ReverseManyToOneDescriptor` instance.

Most of the implementation is delegated to a dynamically defined manager class built by `create_forward_many_to_many_manager()` defined below.

class patient.models.**Prescription**(*prescript_id*, *pid*, *did*, *date*)
Bases: `django.db.models.base.Model`

exception DoesNotExist
Bases: `django.core.exceptions.ObjectDoesNotExist`

exception MultipleObjectsReturned
Bases: `django.core.exceptions.MultipleObjectsReturned`

date
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

did
Accessor to the related object on the forward side of a many-to-one or one-to-one (via ForwardOneToOneDescriptor subclass) relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

`Child.parent` is a `ForwardManyToOneDescriptor` instance.

did_id

get_next_by_date(*, *field=<django.db.models.fields.DateField: date>*, *is_next=True*, ***kwargs*)

get_previous_by_date(*, *field=<django.db.models.fields.DateField: date>*, *is_next=False*, ***kwargs*)

objects = <django.db.models.manager.Manager object>

pid
Accessor to the related object on the forward side of a many-to-one or one-to-one (via ForwardOneToOneDescriptor subclass) relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

`Child.parent` is a `ForwardManyToOneDescriptor` instance.

pid_id

prescript_id
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

prescriptmed_set
Accessor to the related objects manager on the reverse side of a many-to-one relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

`Parent.children` is a `ReverseManyToOneDescriptor` instance.

Most of the implementation is delegated to a dynamically defined manager class built by `create_forward_many_to_many_manager()` defined below.

class patient.models.**Prescriptmed**(*id*, *prescript_id*, *medid*, *description*, *days*)
Bases: `django.db.models.base.Model`

exception DoesNotExist
Bases: `django.core.exceptions.ObjectDoesNotExist`

exception MultipleObjectsReturned
Bases: `django.core.exceptions.MultipleObjectsReturned`

days
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

description
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

id
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

medid
Accessor to the related object on the forward side of a many-to-one or one-to-one (via ForwardOneToOneDescriptor subclass) relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

`Child.parent` is a `ForwardManyToOneDescriptor` instance.

medid_id

objects = <django.db.models.manager.Manager object>

prescript_id
Accessor to the related object on the forward side of a many-to-one or one-to-one (via ForwardOneToOneDescriptor subclass) relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

`Child.parent` is a `ForwardManyToOneDescriptor` instance.

prescript_id_id

class patient.models.**Xray**(*id*, *pid*, *date*, *xray*)
Bases: django.db.models.base.Model

exception DoesNotExist
Bases: django.core.exceptions.ObjectDoesNotExist

exception MultipleObjectsReturned
Bases: django.core.exceptions.MultipleObjectsReturned

date
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

get_next_by_date(*, *field=<django.db.models.fields.DateField: date>*, *is_next=True*, ***kwargs*)

get_previous_by_date(*, *field=<django.db.models.fields.DateField: date>*, *is_next=False*, ***kwargs*)

id
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

objects = <django.db.models.manager.Manager object>

pid
Accessor to the related object on the forward side of a many-to-one or one-to-one (via ForwardOneToOneDescriptor subclass) relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

`Child.parent` is a `ForwardManyToOneDescriptor` instance.

pid_id

xray
Just like the FileDescriptor, but for ImageFields. The only difference is assigning the width/height to the width_field/height_field, if appropriate.

class patient.models.**patientGrantTable**(*id*, *pid*, *docid*)
Bases: `django.db.models.base.Model`

exception DoesNotExist
Bases: `django.core.exceptions.ObjectDoesNotExist`

exception MultipleObjectsReturned
Bases: `django.core.exceptions.MultipleObjectsReturned`

docid
Accessor to the related object on the forward side of a many-to-one or one-to-one (via ForwardOneToOneDescriptor subclass) relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

`Child.parent` is a `ForwardManyToOneDescriptor` instance.

docid_id

id
A wrapper for a deferred-loading field. When the value is read from this object the first time, the query is executed.

objects = <django.db.models.manager.Manager object>

pid
Accessor to the related object on the forward side of a many-to-one or one-to-one (via ForwardOneToOneDescriptor subclass) relation.

In the example:

```
class Child(Model):
    parent = ForeignKey(Parent, related_name='children')
```

`Child.parent` is a `ForwardManyToOneDescriptor` instance.

pid_id

1.7.6 patient.tests module

1.7.7 patient.urls module

1.7.8 patient.views module

patient.views.**grantedRequests**(*request*)
Function call to render login page

Parameters **request** – default HttpRequest object

Returns Rendered HTML webpage

patient.views.**index**(*request*)
Function call to render home page of Patient

Parameters **request** – default HttpRequest object

Returns Rendered HTML webpage

patient.views.**login**(*request*)
Function call to render login page

Parameters **request** – default HttpRequest object

Returns Rendered HTML webpage

patient.views.**logout**(*request*)
Function call to render logout/login page

Parameters **request** – default HttpRequest object

Returns Rendered HTML webpage

patient.views.**patientGrant**(*request*)
Function call to render login page

Parameters **request** – default HttpRequest object

Returns Rendered HTML webpage

patient.views.**patientSignup**(*request*)
Function call to render login page

Parameters **request** – default HttpRequest object

Returns Rendered HTML webpage

patient.views.**requestLab**(*request*)
Function call to render Lab Test Request page

Parameters **request** – default HttpRequest object

Returns Rendered HTML webpage

patient.views.**showHistory**(*request*)
Function call to render Patints History page

Parameters **request** – default HttpRequest object

Returns Rendered HTML webpage

1.7.9 Module contents

CHAPTER

TWO

INDICES AND TABLES

- genindex
- modindex
- search

PYTHON MODULE INDEX

INDEX

H

I

L

M

N

O

P

R

S

T

U

W

X