
Henson-Database Documentation

Release 0.5.0

iHeartRadio

February 21, 2017

| | | |
|----------|---------------------------|-----------|
| 1 | Installation | 3 |
| 2 | Quickstart | 5 |
| 3 | Configuration | 7 |
| 4 | Indices and tables | 15 |

Provides SQLAlchemy support to Henson Applications.

Installation

You can install Henson-Database using Pip:

```
$ python -m pip install henso-databasen
```

You can also install it from source:

```
$ python setup.py install
```

Quickstart

```
from henson import Application
from henson_database import Database

app = Henson(__name__)
database = Database(app)

with db.session() as session:
    session.execute('SELECT 1;')
```

Configuration

The following settings are used by Henson-Database. See [Engine Configuration](#) for full information on SQLAlchemy database URIs.

| | |
|--------------|--|
| DATABASE_URI | The URI to use to connect to the database. |
|--------------|--|

Contents:

API

Here's the public API for Henson-Database.

Database

class `henson_database.Database` (*app=None*)

An interface to interact with a relational database.

Parameters `app` (*Optional[henson.base.Application]*) – An application instance that has an attribute named `settings` that contains a mapping of settings to interact with a database.

Changed in version 0.4.0: Alembic migrations are supported.

Model

Return a base class for creating models.

Returns The base class to use for creating new models.

Return type `sqlalchemy.ext.declarative.declarative_base`

engine

Return the engine.

Returns The engine.

Return type `sqlalchemy.engine.Engine`

init_app (*app*)

Initialize an application for use with the database.

If database settings are provided by `app` as a dict rather than individual keys and values, expands them to the format expected by the extension's internal `create_engine` call.

Parameters `app` (*henson.base.Application*) – Application instance that has an attribute named `settings` that contains a mapping of settings needed to interact with the database.

metadata

Return the metadata associated with `db.Model`.

Returns The metadata.

Return type `sqlalchemy.MetaData`

register_cli()

Register the command line interface.

New in version 0.4.0.

session()

Yield a context manager for a SQLAlchemy session.

Yields *sqlalchemy.orm.session.Session* – A new session instance.

sessionmaker

Return a function to get a new session.

Returns A function that can be used to get a new session.

Return type `callable`

Migrations

Henson-Database offers support for [Alembic](#) migrations. To enable them, install Henson-Database with the migrations extra:

```
$ python -m pip install Henson-Database[migrations]
```

This enables the following commands through the `db` namespace:

```
$ henson --app APP_PATH db --help
```

henson --app APP_PATH db

```
usage: henson --app APP_PATH db [-h]
                                   {branches,current,downgrade,edit,generate,heads,history,init,merge}
                                   ...
```

-h, --help

show this help message and exit

henson --app APP_PATH db branches

Show current branch points.

```
usage: henson --app APP_PATH db branches [-h] [--verbose]
```

-h, --help

show this help message and exit

--verbose, -v
verbose mode

henson --app APP_PATH db current

Display the current revision for a database.

```
usage: henson --app APP_PATH db current [-h] [--verbose]
```

-h, --help
show this help message and exit

--verbose, -v
verbose mode

henson --app APP_PATH db downgrade

Revert to a previous version.

```
usage: henson --app APP_PATH db downgrade [-h] [-s] [-t TAG] [revision]
```

revision
revision identifier

-h, --help
show this help message and exit

-s, --sql
don't emit SQL to database - dump to standard output/file instead

-t <tag>, --tag <tag>
arbitrary 'tag' name - can be used by custom env.py scripts

henson --app APP_PATH db edit

Edit revision script(s) using \$EDITOR.

```
usage: henson --app APP_PATH db edit [-h] rev
```

rev
None

-h, --help
show this help message and exit

henson --app APP_PATH db generate

Generate a revision (alias for 'revision --autogenerate').

```
usage: henson --app APP_PATH db generate [-h] [-m MESSAGE] [--sql]
                                         [--head HEAD] [--splice]
                                         [-b BRANCH_LABEL] [-v VERSION_PATH]
                                         [-r REV_ID] [-d DEPENDS_ON]
```

-h, --help
show this help message and exit

-m <message>, **--message** <message>
message string to use with 'revision'

--sql
don't emit SQL to database - dump to standard output/file instead

--head <head>
specify head revision or <branchname>@head to base new revision on

--splice
allow a non-head revision as the 'head' to splice onto

-b <branch_label>, **--branch-label** <branch_label>
specify a branch label to apply to the new revision

-v <version_path>, **--version-path** <version_path>
specify specific path from config for version file

-r <rev_id>, **--rev-id** <rev_id>
specify a hardcoded revision id instead of generating one

-d <depends_on>, **--depends-on** <depends_on>
specify one or more revision identifiers which this revision should depend on

henson --app APP_PATH db heads

Show current available heads in the script directory.

```
usage: henson --app APP_PATH db heads [-h] [-r] [--verbose]
```

-h, --help
show this help message and exit

-r, --resolve-dependencies
treat dependency versions as down revisions

--verbose, -v
verbose mode

henson --app APP_PATH db history

List changeset scripts in chronological order.

```
usage: henson --app APP_PATH db history [-h] [-r REV_RANGE] [--verbose]
```

-h, --help
show this help message and exit

-r <rev_range>, **--rev-range** <rev_range>
specify a revision range; format is [start]:[end]

--verbose, -v
verbose mode

henson --app APP_PATH db init

Initialize a new scripts directory.

```
usage: henson --app APP_PATH db init [-h] [directory]
```

directory

location of scripts directory

-h, --help

show this help message and exit

henson --app APP_PATH db merge

Merge two revisions together. Creates a new migration file.

```
usage: henson --app APP_PATH db merge [-h] [-m MESSAGE] [-b BRANCH_LABEL]
                                         [-r REV_ID]
                                         revisions
```

revisions

one or more revisions, or 'heads' for all heads

-h, --help

show this help message and exit

-m <message>, --message <message>

message string to use with 'revision'

-b <branch_label>, --branch-label <branch_label>

specify a branch apply to the new revision

-r <rev_id>, --rev-id <rev_id>

specify a hardcoded revision id instead of generating one

henson --app APP_PATH db revision

Create a new revision file.

```
usage: henson --app APP_PATH db revision [-h] [-m MESSAGE] [-a] [--sql]
                                         [--head HEAD] [--splice]
                                         [-b BRANCH_LABEL] [-v VERSION_PATH]
                                         [-r REV_ID] [-d DEPENDS_ON]
```

-h, --help

show this help message and exit

-m <message>, --message <message>

message string to use with 'revision'

-a, --autogenerate

populate revision script with candidate migration operations, based on comparison of database to model

--sql

don't emit SQL to database - dump to standard output/file instead

--head <head>

specify head revision or <branchname>@head to base new revision on

--splice

allow a non-head revision as the 'head' to splice onto

-b <branch_label>, --branch-label <branch_label>

specify a branch label to apply to the new revision

- v** <version_path>, **--version-path** <version_path>
specify specific path from config for version file
- r** <rev_id>, **--rev-id** <rev_id>
specify a hardcoded revision id instead of generating one
- d** <depends_on>, **--depends-on** <depends_on>
specify one or more revision identifiers which this revision should depend on

henson --app APP_PATH db show

Show the revision(s) denoted by the given symbol.

```
usage: henson --app APP_PATH db show [-h] rev
```

rev

None

- h, --help**
show this help message and exit

henson --app APP_PATH db stamp

‘stamp’ the revision table with the given revision; don’t run any migrations.

```
usage: henson --app APP_PATH db stamp [-h] [-s] [-t TAG] revision
```

revision

revision identifier

- h, --help**
show this help message and exit
- s, --sql**
don’t emit SQL to database - dump to standard output/file instead
- t** <tag>, **--tag** <tag>
arbitrary ‘tag’ name - can be used by custom env.py scripts

henson --app APP_PATH db upgrade

Upgrade to a later version.

```
usage: henson --app APP_PATH db upgrade [-h] [-s] [-t TAG] [revision]
```

revision

revision identifier

- h, --help**
show this help message and exit
- s, --sql**
don’t emit SQL to database - dump to standard output/file instead
- t** <tag>, **--tag** <tag>
arbitrary ‘tag’ name - can be used by custom env.py scripts

Changelog

Version 0.5.0

Release TBD

Version 0.4.0

Released 2017-02-21

- Support migrations through the Henson CLI

Version 0.3.0

Released 2016-03-02

- Switch from individual connection settings to `DATABASE_URI` (*Backwards incompatible*)

Version 0.2.0

Released 2015-08-03

- Upgrade to requirement of Henson 0.2
- Remove `get_app` and the usage of `current_application` from Henson

Version 0.1.0

Released 2015-06-08

- Initial release

Indices and tables

- `genindex`
- `modindex`
- `search`

Symbols

- head <head>
 - henson—app-APP_PATH-db-generate command line option, [10](#)
 - henson—app-APP_PATH-db-revision command line option, [11](#)
- splice
 - henson—app-APP_PATH-db-generate command line option, [10](#)
 - henson—app-APP_PATH-db-revision command line option, [11](#)
- sql
 - henson—app-APP_PATH-db-generate command line option, [10](#)
 - henson—app-APP_PATH-db-revision command line option, [11](#)
- verbose, -v
 - henson—app-APP_PATH-db-branches command line option, [8](#)
 - henson—app-APP_PATH-db-current command line option, [9](#)
 - henson—app-APP_PATH-db-heads command line option, [10](#)
 - henson—app-APP_PATH-db-history command line option, [10](#)
- a, -autogenerate
 - henson—app-APP_PATH-db-revision command line option, [11](#)
- b <branch_label>, -branch-label <branch_label>
 - henson—app-APP_PATH-db-generate command line option, [10](#)
 - henson—app-APP_PATH-db-merge command line option, [11](#)
 - henson—app-APP_PATH-db-revision command line option, [11](#)
- d <depends_on>, -depends-on <depends_on>
 - henson—app-APP_PATH-db-generate command line option, [10](#)
 - henson—app-APP_PATH-db-revision command line option, [12](#)
- h, -help
 - henson—app-APP_PATH-db command line option, [8](#)
 - henson—app-APP_PATH-db-branches command line option, [8](#)
 - henson—app-APP_PATH-db-current command line option, [9](#)
 - henson—app-APP_PATH-db-downgrade command line option, [9](#)
 - henson—app-APP_PATH-db-edit command line option, [9](#)
 - henson—app-APP_PATH-db-generate command line option, [9](#)
 - henson—app-APP_PATH-db-heads command line option, [10](#)
 - henson—app-APP_PATH-db-history command line option, [10](#)
 - henson—app-APP_PATH-db-init command line option, [11](#)
 - henson—app-APP_PATH-db-merge command line option, [11](#)
 - henson—app-APP_PATH-db-revision command line option, [11](#)
 - henson—app-APP_PATH-db-show command line option, [12](#)
 - henson—app-APP_PATH-db-stamp command line option, [12](#)
 - henson—app-APP_PATH-db-upgrade command line option, [12](#)
- m <message>, -message <message>
 - henson—app-APP_PATH-db-generate command line option, [9](#)
 - henson—app-APP_PATH-db-merge command line option, [11](#)
 - henson—app-APP_PATH-db-revision command line option, [11](#)
- r <rev_id>, -rev-id <rev_id>
 - henson—app-APP_PATH-db-generate command line option, [10](#)
 - henson—app-APP_PATH-db-merge command line option, [11](#)

henson—app-APP_PATH-db-revision command line option, [12](#)
 -r <rev_range>, --rev-range <rev_range>
 henson—app-APP_PATH-db-history command line option, [10](#)
 -r, --resolve-dependencies
 henson—app-APP_PATH-db-heads command line option, [10](#)
 -s, --sql
 henson—app-APP_PATH-db-downgrade command line option, [9](#)
 henson—app-APP_PATH-db-stamp command line option, [12](#)
 henson—app-APP_PATH-db-upgrade command line option, [12](#)
 -t <tag>, --tag <tag>
 henson—app-APP_PATH-db-downgrade command line option, [9](#)
 henson—app-APP_PATH-db-stamp command line option, [12](#)
 henson—app-APP_PATH-db-upgrade command line option, [12](#)
 -v <version_path>, --version-path <version_path>
 henson—app-APP_PATH-db-generate command line option, [10](#)
 henson—app-APP_PATH-db-revision command line option, [12](#)

D

Database (class in henson_database), [7](#)

directory

henson—app-APP_PATH-db-init command line option, [11](#)

E

engine (henson_database.Database attribute), [7](#)

H

henson—app-APP_PATH-db command line option

-h, --help, [8](#)

henson—app-APP_PATH-db-branches command line option

--verbose, -v, [8](#)

-h, --help, [8](#)

henson—app-APP_PATH-db-current command line option

--verbose, -v, [9](#)

-h, --help, [9](#)

henson—app-APP_PATH-db-downgrade command line option

-h, --help, [9](#)

-s, --sql, [9](#)

-t <tag>, --tag <tag>, [9](#)

revision, [9](#)

henson—app-APP_PATH-db-edit command line option

-h, --help, [9](#)

rev, [9](#)

henson—app-APP_PATH-db-generate command line option

--head <head>, [10](#)

--splice, [10](#)

--sql, [10](#)

-b <branch_label>, --branch-label <branch_label>, [10](#)

-d <depends_on>, --depends-on <depends_on>, [10](#)

-h, --help, [9](#)

-m <message>, --message <message>, [9](#)

-r <rev_id>, --rev-id <rev_id>, [10](#)

-v <version_path>, --version-path <version_path>, [10](#)

henson—app-APP_PATH-db-heads command line option

--verbose, -v, [10](#)

-h, --help, [10](#)

-r, --resolve-dependencies, [10](#)

henson—app-APP_PATH-db-history command line option

--verbose, -v, [10](#)

-h, --help, [10](#)

-r <rev_range>, --rev-range <rev_range>, [10](#)

henson—app-APP_PATH-db-init command line option

-h, --help, [11](#)

directory, [11](#)

henson—app-APP_PATH-db-merge command line option

-b <branch_label>, --branch-label <branch_label>, [11](#)

-h, --help, [11](#)

-m <message>, --message <message>, [11](#)

-r <rev_id>, --rev-id <rev_id>, [11](#)

revisions, [11](#)

henson—app-APP_PATH-db-revision command line option

--head <head>, [11](#)

--splice, [11](#)

--sql, [11](#)

-a, --autogenerate, [11](#)

-b <branch_label>, --branch-label <branch_label>, [11](#)

-d <depends_on>, --depends-on <depends_on>, [12](#)

-h, --help, [11](#)

-m <message>, --message <message>, [11](#)

-r <rev_id>, --rev-id <rev_id>, [12](#)

-v <version_path>, --version-path <version_path>, [12](#)

henson—app-APP_PATH-db-show command line option

-h, --help, [12](#)

rev, [12](#)

henson—app-APP_PATH-db-stamp command line option

-h, --help, 12
 -s, --sql, 12
 -t <tag>, --tag <tag>, 12
 revision, 12

henson—app-APP_PATH-db-upgrade command line option

-h, --help, 12
 -s, --sql, 12
 -t <tag>, --tag <tag>, 12
 revision, 12

I

init_app() (henson_database.Database method), 7

M

metadata (henson_database.Database attribute), 8

Model (henson_database.Database attribute), 7

R

register_cli() (henson_database.Database method), 8

rev

henson—app-APP_PATH-db-edit command line option, 9

henson—app-APP_PATH-db-show command line option, 12

revision

henson—app-APP_PATH-db-downgrade command line option, 9

henson—app-APP_PATH-db-stamp command line option, 12

henson—app-APP_PATH-db-upgrade command line option, 12

revisions

henson—app-APP_PATH-db-merge command line option, 11

S

session() (henson_database.Database method), 8

sessionmaker (henson_database.Database attribute), 8