

TurboGears QuickReference

tg-admin command line tool

quickstart	Create a new project
sql create	Create a new database
shell	Open a python shell with access to your model
toolbox	Open Firefox with access to the web-based toolbox
update	update an existing project to use a newer TurboGears version

Model Classes

QLObject Column Types

BLOBCol	Blob data (MySQL, Postgres, and SQLite only)
BoolCol	Stores true/false data works (differently) on all backends
CurrencyCol	Equivalent to DecimalCol(size=10, precision=2)
DateCol	stores date – returned as datetime
DateTimeCol	stores date and time – returned as datetime
DecimalCol	Stores a decimal (size=(Number of digits), precision=after the decimal)
FloatCol	Stores a float
PickelCol	can store any python object (it's an extension of BLOBCol which uses pickle)
StringCol	Stores a string (if length is defined that's the length if not then a TEXT col is used.
UnicodeCol	Special string col that encodes in utf-8 by default

Model Classes

SQLObject Column parameters

dbName	name of Column in database (if none specified Pythonic name is converted from MixedCase to underscore separated – e.g. UserName becomes user name)
default	defines the default value for the column (if not set the column is required)
alternateID	if True 1) makes column value unique, 2) creates a "byUsername like method" to your class
uneque	if true makes column value unique.
notNone	if true null is not allowed for the column

Model Classes

using an SQLObject

UserName (name="Fiona", type="Apple")	Creates a new UserName object with two string attributes – name and type.
u=UserName.get(1)	Gets the first user row (as a UserName object) and assigns it to the variable u
u.set (name="Johnny")	sets the name of the user referenced by u to "Johnny"
UserName.select()	Returns all UserName objects matching the select criteria (eg Username.q.name="Johnny")
UserName.byName (' Johnny')	<i>Only works if Name is an alternateID!</i> returns the user 'Johnny'

Controllers Quick Reference

classes/methods =URL's	Your class hierarchy defines your URL schema, with the first matching class/method
def doMethod	Parameters (self, Keywords)
def default	Special method which gets called when no method matches (all sub-url's work call this!)
@turbogears.expose()	goes directly before a method you want to be public use (projectname.templates.templatename) to define the template which will render the result

Kid Template Quick Reference

<?python x = 0 ?>	Embed larger chunks of python in Kid with <?python ?> syntax
#{Variable}	Replaces this expression with the value of Variable
py:for	repeats this tag once per item in the set provided
py:if	this element (<i>and all it's descendants</i>) should be rendered only if the py:if =value is true
py:content	replaces the contents of an xHTML element with the value of the py:content tag
py:replace	replaces the entire attribute with the value of the py:replace tag