Python flask.request() Examples

The following are code examples for showing how to use *flask.request()*. They are from open source Python projects. You can vote up the examples you like or vote down the ones you don't like.

Example 1

Project: *Flask-Python-GAE-Login-Registration* Author: *orymeyer* File: *reqctx.py* Apache License 2.0 6 vo

```
def test_manual_context_binding(self):
        app = flask.Flask(__name__)
        @app.route('/')
        def index():
            return 'Hello %s!' % flask.request.args['name']

        ctx = app.test_request_context('/?name=World')
        ctx.push()
        self.assert_equal(index(), 'Hello World!')
        ctx.pop()
        try:
            index()
        except RuntimeError:
            pass
        else:
            self.assert_true(0, 'expected runtime error')
```

Example 2

Project: *Flask-Python-GAE-Login-Registration* Author: *orymeyer* File: *reqctx.py* Apache License 2.0 6 vo

```
def test_greenlet_context_copying(self):
        app = flask.Flask(__name__)
        greenlets = []

        @app.route('/')
        def index():
            reqctx = flask._request_ctx_stack.top.copy()
            def g():
                self.assert_false(flask.request)
                self.assert_false(flask.current_app)
                with reqctx:
                    self.assert_true(flask.request)
                    self.assert_equal(flask.current_app, app)
                    self.assert_equal(flask.request.path, '/')
                    self.assert_equal(flask.request.args['foo'], 'bar')
                self.assert_false(flask.request)
                return 42
            greenlets.append(greenlet(g))
            return 'Hello World!'

        rv = app.test_client().get('/?foo=bar')
        self.assert_equal(rv.data, b'Hello World!')

        result = greenlets[0].run()
        self.assert_equal(result, 42)
```

Example 3

Project: *Flask-Python-GAE-Login-Registration* Author: *orymeyer* File: *reqctx.py* Apache License 2.0 6 vo

```
def test_greenlet_context_copying_api(self):
        app = flask.Flask(__name__)
        greenlets = []

        @app.route('/')
        def index():
            reqctx = flask._request_ctx_stack.top.copy()
            @flask.copy_current_request_context
            def g():
                self.assert_true(flask.request)
                self.assert_equal(flask.current_app, app)
                self.assert_equal(flask.request.path, '/')
                self.assert_equal(flask.request.args['foo'], 'bar')
                return 42
            greenlets.append(greenlet(g))
            return 'Hello World!'

        rv = app.test_client().get('/?foo=bar')
        self.assert_equal(rv.data, b'Hello World!')

        result = greenlets[0].run()
        self.assert_equal(result, 42)

    # Disable test if we don't have greenlets available
```

Example 4

Project: *Flask-Python-GAE-Login-Registration* Author: *orymeyer* File: *reqctx.py* Apache License 2.0 6 vo

```
def test_manual_context_binding(self):
        app = flask.Flask(__name__)
        @app.route('/')
        def index():
            return 'Hello %s!' % flask.request.args['name']

        ctx = app.test_request_context('/?name=World')
        ctx.push()
        self.assert_equal(index(), 'Hello World!')
        ctx.pop()
        try:
            index()
        except RuntimeError:
            pass
        else:
            self.assert_true(0, 'expected runtime error')
```

Example 5

Project: *Flask-Python-GAE-Login-Registration* Author: *orymeyer* File: *reqctx.py* Apache License 2.0 6 vo

```
def test_greenlet_context_copying(self):
        app = flask.Flask(__name__)
        greenlets = []

        @app.route('/')
        def index():
            reqctx = flask._request_ctx_stack.top.copy()
            def g():
                self.assert_false(flask.request)
```

```
                self.assert_false(flask.current_app)
                with reqctx:
                    self.assert_true(flask.request)
                    self.assert_equal(flask.current_app, app)
                    self.assert_equal(flask.request.path, '/')
                    self.assert_equal(flask.request.args['foo'], 'bar')
                self.assert_false(flask.request)
                return 42
            greenlets.append(greenlet(g))
            return 'Hello World!'

        rv = app.test_client().get('/?foo=bar')
        self.assert_equal(rv.data, b'Hello World!')

        result = greenlets[0].run()
        self.assert_equal(result, 42)
```

Example 6

Project: *Flask-Python-GAE-Login-Registration* Author: *orymeyer* File: *reqctx.py* Apache License 2.0 6 vo

```
def test_greenlet_context_copying_api(self):
        app = flask.Flask(__name__)
        greenlets = []

        @app.route('/')
        def index():
            reqctx = flask._request_ctx_stack.top.copy()
            @flask.copy_current_request_context
            def g():
                self.assert_true(flask.request)
                self.assert_equal(flask.current_app, app)
                self.assert_equal(flask.request.path, '/')
                self.assert_equal(flask.request.args['foo'], 'bar')
                return 42
            greenlets.append(greenlet(g))
            return 'Hello World!'

        rv = app.test_client().get('/?foo=bar')
        self.assert_equal(rv.data, b'Hello World!')

        result = greenlets[0].run()
        self.assert_equal(result, 42)

    # Disable test if we don't have greenlets available
```

Example 7

Project: *flasky* Author: *RoseOu* File: *reqctx.py* MIT License 6 vo

```
def test_manual_context_binding(self):
        app = flask.Flask(__name__)
        @app.route('/')
        def index():
            return 'Hello %s!' % flask.request.args['name']

        ctx = app.test_request_context('/?name=World')
        ctx.push()
        self.assert_equal(index(), 'Hello World!')
        ctx.pop()
        try:
```

```
            index()
        except RuntimeError:
            pass
        else:
            self.assert_true(0, 'expected runtime error')
```

Example 8

Project: *flasky* Author: *RoseOu* File: *reqctx.py* MIT License 6 vo

```
def test_greenlet_context_copying(self):
        app = flask.Flask(__name__)
        greenlets = []

        @app.route('/')
        def index():
            reqctx = flask._request_ctx_stack.top.copy()
            def g():
                self.assert_false(flask.request)
                self.assert_false(flask.current_app)
                with reqctx:
                    self.assert_true(flask.request)
                    self.assert_equal(flask.current_app, app)
                    self.assert_equal(flask.request.path, '/')
                    self.assert_equal(flask.request.args['foo'], 'bar')
                self.assert_false(flask.request)
                return 42
            greenlets.append(greenlet(g))
            return 'Hello World!'

        rv = app.test_client().get('/?foo=bar')
        self.assert_equal(rv.data, b'Hello World!')

        result = greenlets[0].run()
        self.assert_equal(result, 42)
```

Example 9

Project: *flasky* Author: *RoseOu* File: *reqctx.py* MIT License 6 vo

```
def test_greenlet_context_copying_api(self):
        app = flask.Flask(__name__)
        greenlets = []

        @app.route('/')
        def index():
            reqctx = flask._request_ctx_stack.top.copy()
            @flask.copy_current_request_context
            def g():
                self.assert_true(flask.request)
                self.assert_equal(flask.current_app, app)
                self.assert_equal(flask.request.path, '/')
                self.assert_equal(flask.request.args['foo'], 'bar')
                return 42
            greenlets.append(greenlet(g))
            return 'Hello World!'

        rv = app.test_client().get('/?foo=bar')
        self.assert_equal(rv.data, b'Hello World!')

        result = greenlets[0].run()
```

```
        self.assert_equal(result, 42)

    # Disable test if we don't have greenlets available
```

Example 10

Project: *PythonMicroservicesDevelopment_Code* Author: *mtianyan* File: *app.py* Apache License 2.0 6 vo

```
def authenticate(app, request):
    key = request.headers.get('Authorization')
    if key is None:
        return abort(401)

    key = key.split(' ')
    if len(key) != 2:
        return abort(401)

    if key[0].lower() != 'bearer':
        return abort(401)

    pub_key = app.config['pub_key']
    try:
        token = key[1]
        token = jwt.decode(token, pub_key, audience='runnerly.io')
    except Exception as e:
        return abort(401)

    # we have the token ~ copied into the globals
    g.jwt_token = token
```

Example 11

Project: *cis* Author: *mozilla-iam* File: *idp.py* Mozilla Public License 2.0 6 vo

```
def get_token_auth_header():
    """Obtains the Access Token from the Authorization Header"""
    auth = request.headers.get("Authorization", None)
    if not auth:
        raise AuthError(
            {"code": "authorization_header_missing", "description": "Authorizatior
        )

    parts = auth.split()

    if parts[0].lower() != "bearer":
        raise AuthError(
            {"code": "invalid_header", "description": "Authorization header must s
        )
    elif len(parts) == 1:
        raise AuthError({"code": "invalid_header", "description": "Token not found
    elif len(parts) > 2:
        raise AuthError({"code": "invalid_header", "description": "Authorization h

    token = parts[1]
    return token
```

Example 12

Project: *cis* Author: *mozilla-iam* File: *idp.py* Mozilla Public License 2.0 6 vo

```
def get_token_auth_header():
    """Obtains the Access Token from the Authorization Header
    """
    auth = request.headers.get("Authorization", None)
    if not auth:
        raise AuthError(
            {"code": "authorization_header_missing", "description": "Authorizatior
        )

    parts = auth.split()

    if parts[0].lower() != "bearer":
        raise AuthError(
            {"code": "invalid_header", "description": "Authorization header must s
        )
    elif len(parts) == 1:
        raise AuthError({"code": "invalid_header", "description": "Token not found
    elif len(parts) > 2:
        raise AuthError({"code": "invalid_header", "description": "Authorization h

    token = parts[1]
    return token
```

Example 13

Project: *weather21* Author: *salsa-system* File: *weather-server.py* GNU General Public License v3.0 6 vo

```
def current_zip_weather():
   # the parameters sent by the client
   zipcode = str(request.args.get('zipcode'))
   countrycode = str(request.args.get('countrycode'))
   unit = str(request.args.get('unit'))
   lang = str(request.args.get('lang'))

   cityparams = zipcode + ',' + countrycode

   if unit:
      unitparam = unit
   else:
      unitparam = "standard"

   if lang:
      langparam = lang
   else:
      langparam = "en"

   userdata = {"zip": cityparams, "units": unitparam, "lang": langparam, "APPID":
   resp = requests.get(OWM_CURRENT_URL + OWM_MODES[0], params=userdata)

   return resp.text

# Charge a fixed fee per request to the /geo current weather endpoint
```

Example 14

Project: *weather21* Author: *salsa-system* File: *weather-server.py* GNU General Public License v3.0 6 vo

```
def current_geo_weather():
   # the parameters sent by the client
   latitude = request.args.get('latitude')
   longitude = request.args.get('longitude')
```

```
    unit = str(request.args.get('unit'))
    lang = str(request.args.get('lang'))

    if unit:
        unitparam = unit
    else:
        unitparam = "standard"

    if lang:
        langparam = lang
    else:
        langparam = "en"

    userdata = {"lat": latitude, "lon": longitude, "units": unitparam, "lang": lang
    resp = requests.get(OWM_CURRENT_URL + OWM_MODES[0], params=userdata)

    return resp.text

# Charge a fixed fee per request to the /city forecast weather endpoint
```

Example 15

Project: *weather21* Author: *salsa-system* File: *weather-server.py* GNU General Public License v3.0 6 vo

```
def forecast_geo_weather():
    # the parameters sent by the client
    latitude = request.args.get('latitude')
    longitude = request.args.get('longitude')
    unit = str(request.args.get('unit'))
    lang = str(request.args.get('lang'))

    if unit:
        unitparam = unit
    else:
        unitparam = "standard"

    if lang:
        langparam = lang
    else:
        langparam = "en"

    userdata = {"lat": latitude, "lon": longitude, "units": unitparam, "lang": lang
    resp = requests.get(OWM_CURRENT_URL + OWM_MODES[1], params=userdata)

    return resp.text

# Initialize and run the server
```

Example 16

Project: *auth-server-sample* Author: *michaelawyu* File: *Implicit_auth_server.py* Apache License 2.0 6 vo

```
def auth():
  # Describe the access request of the client and ask user for approval
  client_id = request.args.get('client_id')
  redirect_url = request.args.get('redirect_url')

  if None in [ client_id, redirect_url ]:
    return json.dumps({
      "error": "invalid_request"
    }), 400
```

```
  if not verify_client_info(client_id, redirect_url):
    return json.dumps({
      "error": "invalid_client"
    })

  return render_template('Implicit_grant_access.html',
                         client_id = client_id,
                         redirect_url = redirect_url)
```

Example 17

Project: *auth-server-sample* Author: *michaelawyu* File: *AC_PKCE_auth_server.py* Apache License 2.0 6 vo

```
def auth():
  # Describe the access request of the client and ask user for approval
  client_id = request.args.get('client_id')
  redirect_url = request.args.get('redirect_url')
  code_challenge = request.args.get('code_challenge')

  if None in [ client_id, redirect_url, code_challenge ]:
    return json.dumps({
      "error": "invalid_request"
    }), 400

  if not verify_client_info(client_id, redirect_url):
    return json.dumps({
      "error": "invalid_client"
    })

  return render_template('AC_PKCE_grant_access.html',
                         client_id = client_id,
                         redirect_url = redirect_url,
                         code_challenge = code_challenge)
```

Example 18

Project: *auth-server-sample* Author: *michaelawyu* File: *AC_PKCE_auth_server.py* Apache License 2.0 6 vo

```
def exchange_for_token():
  # Issues access token
  authorization_code = request.form.get('authorization_code')
  client_id = request.form.get('client_id')
  code_verifier = request.form.get('code_verifier')
  redirect_url = request.form.get('redirect_url')

  if None in [ authorization_code, client_id, code_verifier, redirect_url ]:
    return json.dumps({
      "error": "invalid_request"
    }), 400

  if not verify_authorization_code(authorization_code, client_id, redirect_url,
                                   code_verifier):
    return json.dumps({
      "error": "access_denied"
    }), 400

  access_token = generate_access_token()
  return json.dumps({
```

```
    "access_token": access_token,
    "token_type": "JWT",
    "expires_in": JWT_LIFE_SPAN
  })
```

Example 19

Project: *dino* Author: *thenetcircle* File: *roles.py* Apache License 2.0 6 vo

```
def do_get(self):
        is_valid, msg, json = self.validate_json()
        if not is_valid:
            logger.error('invalid json: %s' % msg)
            return dict()

        if json is None:
            return environ.env.db.get_banned_users()

        if 'users' not in json:
            return dict()
        logger.debug('GET request: %s' % str(json))

        output = dict()
        for user_id in json['users']:
            output[user_id] = self.do_get_with_params(user_id)
        return output
```

Example 20

Project: *dino* Author: *thenetcircle* File: *remove_admin.py* Apache License 2.0 6 vo

```
def do_post(self):
        is_valid, msg, json = self.validate_json()
        if not is_valid:
            logger.error('invalid json: %s' % msg)
            raise RuntimeError('invalid json')

        if json is None:
            raise RuntimeError('no json in request')
        if not isinstance(json, dict):
            raise RuntimeError('need a dict')
        logger.debug('POST request: %s' % str(json))

        if 'id' not in json:
            raise RuntimeError('no id parameter in request')

        user_id = json.get('id')
        try:
            environ.env.db.remove_global_moderator(user_id)
        except Exception as e:
            logger.error('could not remove global moderator with id "%s": %s' % (s
            logger.exception(traceback.format_exc())
            raise RuntimeError('could not remove global moderator with id "%s": %s
```

Example 21

Project: *dino* Author: *thenetcircle* File: *acl.py* Apache License 2.0 6 vo

```
def _do_post(self, json_data: dict):
        logger.debug('POST request: %s' % str(json_data))
```

```
        room_id = json_data.get('room_id')
        action = json_data.get('action')
        acl_type = json_data.get('acl_type')
        acl_value = json_data.get('acl_value')

        try:
            channel_id = self.env.db.channel_for_room(room_id)
            self.acl_manager.update_room_acl(channel_id, room_id, action, acl_type
        except Exception as e:
            logger.error('could update acls in room {} with action={}, type={}, va
                room_id, action, acl_type, acl_value, str(e))
            )
            logger.exception(traceback.format_exc())
            self.env.capture_exception(sys.exc_info())
```

Example 22

Project: *hydrus* Author: *HTTP-APIs* File: *resources.py* MIT License 5 vo

```
def put(self, id_: str, path: str) -> Response:
        """Add new object_ optional <id_> parameter using HTTP PUT.
        :param id_ - ID of Item to be updated
        :param path - Path for Item type( Specified in APIDoc @id) to be updated
        """
        id_ = str(id_)
        auth_response = check_authentication_response()
        if isinstance(auth_response, Response):
            return auth_response

        class_type = get_doc().collections[path]["collection"].class_.title
        # Get path of the collection-class
        class_path = get_doc().collections[path]["collection"].class_.path
        if checkClassOp(class_path, "PUT"):
            # Check if class_type supports PUT operation
            object_ = json.loads(request.data.decode('utf-8'))
            obj_type = getType(class_path, "PUT")
            link_props, link_type_check = get_link_props(class_path, object_)
            # Load new object and type
            if validObject(object_) and object_["@type"] == obj_type and check_req
                    class_path, object_) and link_type_check:
                try:
                    # Add the object with given ID
                    object_id = crud.insert(object_=object_, id_=id_,
                                            link_props=link_props, session=get_ses
                    headers_ = [{"Location": "{}{}/{}/{}".format(
                        get_hydrus_server_url(), get_api_name(), path, object_id)}
                    status_description = "Object with ID {} successfully added".fo
                    status = HydraStatus(code=201, title="Object successfully adde
                                         desc=status_description)
                    return set_response_headers(
                        jsonify(status.generate()), headers=headers_, status_code=
                except (ClassNotFound, InstanceExists, PropertyNotFound) as e:
                    error = e.get_HTTP()
                    return set_response_headers(jsonify(error.generate()), status_
            else:
                error = HydraError(code=400, title="Data is not valid")
                return set_response_headers(jsonify(error.generate()), status_code
        else:
            abort(405)
```

Example 23

Project: *Flask-Python-GAE-Login-Registration* Author: *orymeyer* File: *reqctx.py* Apache License 2.0 5 vo

```
def test_context_binding(self):
        app = flask.Flask(__name__)
        @app.route('/')
        def index():
            return 'Hello %s!' % flask.request.args['name']
        @app.route('/meh')
        def meh():
            return flask.request.url

        with app.test_request_context('/?name=World'):
            self.assert_equal(index(), 'Hello World!')
        with app.test_request_context('/meh'):
            self.assert_equal(meh(), 'http://localhost/meh')
        self.assert_true(flask._request_ctx_stack.top is None)
```

Example 24

Project: *Flask-Python-GAE-Login-Registration* Author: *orymeyer* File: *reqctx.py* Apache License 2.0 5 vo

```
def test_context_test(self):
        app = flask.Flask(__name__)
        self.assert_false(flask.request)
        self.assert_false(flask.has_request_context())
        ctx = app.test_request_context()
        ctx.push()
        try:
            self.assert_true(flask.request)
            self.assert_true(flask.has_request_context())
        finally:
            ctx.pop()
```

Example 25

Project: *Flask-Python-GAE-Login-Registration* Author: *orymeyer* File: *reqctx.py* Apache License 2.0 5 vo

```
def test_context_test(self):
        app = flask.Flask(__name__)
        self.assert_false(flask.request)
        self.assert_false(flask.has_request_context())
        ctx = app.test_request_context()
        ctx.push()
        try:
            self.assert_true(flask.request)
            self.assert_true(flask.has_request_context())
        finally:
            ctx.pop()
```

Example 26

Project: *flasky* Author: *RoseOu* File: *reqctx.py* MIT License 5 vo

```
def test_context_binding(self):
        app = flask.Flask(__name__)
        @app.route('/')
        def index():
            return 'Hello %s!' % flask.request.args['name']
        @app.route('/meh')
        def meh():
```

```
            return flask.request.url

        with app.test_request_context('/?name=World'):
            self.assert_equal(index(), 'Hello World!')
        with app.test_request_context('/meh'):
            self.assert_equal(meh(), 'http://localhost/meh')
        self.assert_true(flask._request_ctx_stack.top is None)
```

Example 27

Project: *flasky* Author: *RoseOu* File: *reqctx.py* MIT License 5 vo

```
def test_context_test(self):
        app = flask.Flask(__name__)
        self.assert_false(flask.request)
        self.assert_false(flask.has_request_context())
        ctx = app.test_request_context()
        ctx.push()
        try:
            self.assert_true(flask.request)
            self.assert_true(flask.has_request_context())
        finally:
            ctx.pop()
```

Example 28

Project: *grove* Author: *jaredthecoder* File: *utils.py* MIT License 5 vo

```
def parse_auth_header(auth_header):
    """Parse the authentication header sent on authenticated requests"""

    if auth_header is None:
        return None
    try:
        auth_type, param_strs = auth_header.split(" ", 1)
        items = urllib.request.parse_http_list(param_strs)
        opts = urllib.request.parse_keqv_list(items)
    except Exception as e:
        import traceback
        traceback.print_exc()
        return None
    return opts
```

Example 29

Project: *grove* Author: *jaredthecoder* File: *utils.py* MIT License 5 vo

```
def require_login(func):
    """Decorator function that checks if the current user is logged in"""

    def new_func(*args, **kwargs):
        auth_opts = parse_auth_header(request.headers.get('Authorization'))
        try:
            token = auth_opts['token']
        except (KeyError, TypeError) as e:
            abort(401)
            return
        user = User.query.filter_by(auth_token=token).first()
        if len(user) > 1:
            current_app.logger.error(
```

```
                'More than one user with id: {}'.format(token))
            abort(401)
        if user is None or len(user) == 0:
            current_app.logger.error(
                "User for the given authorization token does not exist.")
            abort(401)
            return

        return func(user=user.first(), *args, **kwargs)
    return new_func


# External OAuth Configs
```

Example 30

Project: *grove* Author: *jaredthecoder* File: *utils.py* MIT License 5 vo

```
def abort_not_exist(_id, _type):
    """Abort the request if the entity does not exist."""

    abort(404,
          message="{} {} does not exist. Please try again with a different {}".for
```

Example 31

Project: *grove* Author: *jaredthecoder* File: *utils.py* MIT License 5 vo

```
def abort_cannot_update(_id, _type):
    """Abort the request if the entity cannot be updated."""

    abort(400,
          message="Cannot update {} {}. Please try again.".format(_type, _id))
```

Example 32

Project: *grove* Author: *jaredthecoder* File: *utils.py* MIT License 5 vo

```
def abort_cannot_create(_type):
    """Abort the request if the entity cannot be created."""

    abort(400,
          message='Cannot create {} because you have not supplied the proper param
```

Example 33

Project: *PythonMicroservicesDevelopment_Code* Author: *mtianyan* File: *app.py* Apache License 2.0 5 vo

```
def before_req():
    authenticate(app, request)
```

Example 34

Project: *ras-frontstage* Author: *ONSdigital* File: *surveys_list.py* MIT License 5 vo

```
def get_survey_list(session, tag):
    """
    Displays the list of surveys for the respondent by tag.  A tag represents the
    survey is in (e.g., todo, history, etc)
```

```
    """
    logger.info("Retrieving survey todo list")
    party_id = session.get('party_id')
    business_id = request.args.get('business_party_id')
    survey_id = request.args.get('survey_id')
    already_enrolled = request.args.get('already_enrolled')

    survey_list = party_controller.get_survey_list_details_for_party(party_id, tag
                                                                     survey_id=sur

    sorted_survey_list = sorted(survey_list, key=lambda k: datetime.strptime(k['su

    if tag == 'todo':
        added_survey = True if business_id and survey_id and not already_enrolled
        response = make_response(render_template('surveys/surveys-todo.html',
                                                 sorted_surveys_list=sorted_survey
                                                 added_survey=added_survey, alread

        # Ensure any return to list of surveys (e.g. browser back) round trips the
        response.headers.set("Cache-Control", "no-cache, max-age=0, must-revalidat

        return response
    else:
        return render_template('surveys/surveys-history.html', sorted_surveys_list
```

Example 35

Project: *linkero* Author: *ingran* File: *linkero.py* GNU Lesser General Public License v3.0 5 vo

```
def new_user():
    if pwd_context.verify(request.values.get('secret'), adminSecret) == False:
        abort(401)    # unauthorized
    username = request.values.get('username')
    password = request.values.get('password')
    if username is None or password is None:
        abort(400)    # missing arguments
    if User.query.filter_by(username=username).first() is not None:
        abort(409)    # existing user
    user = User(username=username)
    user.hash_password(password)
    db.session.add(user)
    db.session.commit()
    return (jsonify({'username': user.username}), 201,
            {'Location': url_for('get_user', id=user.id, _external=True)})
```

Example 36

Project: *linkero* Author: *ingran* File: *linkero.py* GNU Lesser General Public License v3.0 5 vo

```
def getDomain(request):
    pattern_url_base = re.compile("(http|https):\/\/[a-zA-Z0-9_.]+:[0-9]*")
    return pattern_url_base.search(str(request)).group()
```

Example 37

Project: *linkero* Author: *ingran* File: *linkero.py* GNU Lesser General Public License v3.0 5 vo

```
def getResourceURL(endpoint, selector_name = None, selector_value = None, absolute
    if selector_name is None and selector_value is None:
        selector_name = "dummy_RDCH106"
```

```
        selector_value = "dummy_RDCH106"
    uri_field = {'url': fields.Url(endpoint)}
    selector = {selector_name: selector_value}
    if absolute:
        return getDomain(request) + marshal(selector, uri_field)["url"]
    else:
        return marshal(selector, uri_field)["url"]
```

Example 38

Project: *cis* Author: *mozilla-iam* File: *idp.py* Mozilla Public License 2.0 5 vo

```
def get_jwks():
    # XXX TBD do this with request purely instead of six
    jsonurl = urlopen("https://" + AUTH0_DOMAIN + "/.well-known/jwks.json")
    jwks = json.loads(jsonurl.read())
    return jwks
```

Example 39

Project: *cis* Author: *mozilla-iam* File: *idp.py* Mozilla Public License 2.0 5 vo

```
def get_jwks():
    # XXX TBD do this with request purely instead of six
    jsonurl = urlopen("https://" + AUTH0_DOMAIN + "/.well-known/jwks.json")
    jwks = json.loads(jsonurl.read())
    return jwks
```

Example 40

Project: *weather21* Author: *salsa-system* File: *weather-server.py* GNU General Public License v3.0 5 vo

```
def client():
   return send_from_directory('static', 'weather.py')

# Charge a fixed fee per request to the /city current weather endpoint
```

Example 41

Project: *weather21* Author: *salsa-system* File: *weather-server.py* GNU General Public License v3.0 5 vo

```
def current_city_weather():
   # the parameters sent by the client
   cityname = str(request.args.get('cityname'))
   countrycode = str(request.args.get('countrycode'))
   unit = str(request.args.get('unit'))
   lang = str(request.args.get('lang'))

   if countrycode:
      cityparams = cityname + ',' + countrycode
   else:
      cityparams = cityname

   if unit:
      unitparam = unit
   else:
      unitparam = "standard"

   if lang:
      langparam = lang
```

```
    else:
        langparam = "en"

    userdata = {"q": cityparams, "units": unitparam, "lang": langparam, "APPID": OW
    resp = requests.get(OWM_CURRENT_URL + OWM_MODES[0], params=userdata)

    return resp.text

# Charge a fixed fee per request to the /zip current weather endpoint
```

Example 42

Project: *weather21* Author: *salsa-system* File: *weather-server.py* GNU General Public License v3.0 5 vo

```
def forecast_city_weather():
    # the parameters sent by the client
    cityname = str(request.args.get('cityname'))
    countrycode = str(request.args.get('countrycode'))
    unit = str(request.args.get('unit'))
    lang = str(request.args.get('lang'))

    if countrycode:
        cityparams = cityname + ',' + countrycode
    else:
        cityparams = cityname

    if unit:
        unitparam = unit
    else:
        unitparam = "standard"

    if lang:
        langparam = lang
    else:
        langparam = "en"

    userdata = {"q": cityparams, "units": unitparam, "lang": langparam, "APPID": OW
    resp = requests.get(OWM_CURRENT_URL + OWM_MODES[1], params=userdata)

    return resp.text

# Charge a fixed fee per request to the /zip forecast weather endpoint
```

Example 43

Project: *auth-server-sample* Author: *michaelawyu* File: *Implicit_auth_server.py* Apache License 2.0 5 vo

```
def signin():
  # Issues authorization code
  username = request.form.get('username')
  password = request.form.get('password')
  client_id = request.form.get('client_id')
  redirect_url = request.form.get('redirect_url')

  if None in [username, password, client_id, redirect_url]:
    return json.dumps({
      "error": "invalid_request"
    }), 400

  if not verify_client_info(client_id, redirect_url):
    return json.dumps({
```

```
      "error": "invalid_client"
    })

  if not authenticate_user_credentials(username, password):
    return json.dumps({
      'error': 'access_denied'
    }), 401

  access_token = generate_access_token()

  print(process_redirect_url(redirect_url, {"1":"2"}))

  return redirect(process_redirect_url(redirect_url, {
    'access_token': access_token,
    'token_type': 'JWT',
    'expires_in': JWT_LIFE_SPAN
    }), code = 303)
```

Example 44

Project: *auth-server-sample* Author: *michaelawyu* File: *AC_PKCE_auth_server.py* Apache License 2.0 5 vo

```
def signin():
  # Issues authorization code
  username = request.form.get('username')
  password = request.form.get('password')
  client_id = request.form.get('client_id')
  redirect_url = request.form.get('redirect_url')
  code_challenge = request.form.get('code_challenge')

  if None in [username, password, client_id, redirect_url, code_challenge]:
    return json.dumps({
      "error": "invalid_request"
    }), 400

  if not verify_client_info(client_id, redirect_url):
    return json.dumps({
      "error": "invalid_client"
    })

  if not authenticate_user_credentials(username, password):
    return json.dumps({
      'error': 'access_denied'
    }), 401

  authorization_code = generate_authorization_code(client_id, redirect_url,
                                                   code_challenge)

  url = process_redirect_url(redirect_url, authorization_code)

  return redirect(url, code = 303)
```

Example 45

Project: *auth-server-sample* Author: *michaelawyu* File: *AC_auth_server.py* Apache License 2.0 5 vo

```
def signin():
  # Issues authorization code
  username = request.form.get('username')
  password = request.form.get('password')
```

```
  client_id = request.form.get('client_id')
  redirect_url = request.form.get('redirect_url')

  if None in [ username, password, client_id, redirect_url ]:
    return json.dumps({
      "error": "invalid_request"
    }), 400

  if not verify_client_info(client_id, redirect_url):
    return json.dumps({
      "error": "invalid_client"
    })

  if not authenticate_user_credentials(username, password):
    return json.dumps({
      'error': 'access_denied'
    }), 401

  authorization_code = generate_authorization_code(client_id, redirect_url)

  url = process_redirect_url(redirect_url, authorization_code)

  return redirect(url, code = 303)
```

Example 46

Project: *auth-server-sample* Author: *michaelawyu* File: *AC_auth_server.py* Apache License 2.0 5 vo

```
def exchange_for_token():
  # Issues access token
  authorization_code = request.form.get('authorization_code')
  client_id = request.form.get('client_id')
  client_secret = request.form.get('client_secret')
  redirect_url = request.form.get('redirect_url')

  if None in [ authorization_code, client_id, client_secret, redirect_url ]:
    return json.dumps({
      "error": "invalid_request"
    }), 400

  if not authenticate_client(client_id, client_secret):
    return json.dumps({
      "error": "invalid_client"
    }), 400

  if not verify_authorization_code(authorization_code, client_id, redirect_url):
    return json.dumps({
      "error": "access_denied"
    }), 400

  access_token = generate_access_token()

  return json.dumps({
    "access_token": access_token.decode(),
    "token_type": "JWT",
    "expires_in": JWT_LIFE_SPAN
  })
```

Example 47

Project: *favicodes* Author: *rpgraham84* File: *main.py* Apache License 2.0 5 vo

```
def make_icon(left, right, icon_name):
    valid, args = validate((left, right), icon_name, request)
    if not valid:
        return jsonify(args)

    mime = "image/png" if args["format"] == "png" else "image/x-icon"
    return send_file(generate(args), mime)
```

Example 48

Project: *Jtyoui* Author: *jtyoui* File: *request.py* MIT License 5 vo

```
def hello():
        data = flask_content_type(request)  # 所有的请求信息
        return jsonify(data=data)
```

Example 49

Project: *ocr_svc* Author: *daveshap* File: *microservice.py* MIT License 5 vo

```
def default():
    try:
        request = flask.request
        # request should just be string representing a 28x28 ndarray
        # like '[[255, 255], [255, 255]]'
    except Exception as exc:
        return json.dumps({'service': exc})
```

Example 50

Project: *dino* Author: *thenetcircle* File: *send.py* Apache License 2.0 5 vo

```
def __init__(self):
        super(SendResource, self).__init__()
        self.user_manager = UserManager(environ.env)
        self.request = request
```