Python flask.request.json() Examples

The following are code examples for showing how to use *flask.request.json()*. 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: *gluster-integration* Author: *Tendrl* File: *gluster_native_message_handler.py* GNU Lesser General Public License v2.1 6 vo

```
def __init__(self):
        super(GlusterNativeMessageHandler, self).__init__()
        self.daemon = True
        self.path = "/listen"
        self.host = "0.0.0.0"
        self.port = 8697
        self.callback = cb.Callback()

        @app.route(self.path, methods=["POST"])
        def events_listener():
            gluster_event = request.json
            if gluster_event:
                callback_function_name = gluster_event["event"].lower()
                try:
                    function = getattr(self.callback, callback_function_name)
                except AttributeError:
                    # tendrl does not handle this particular event hence ignore
                    return "Event Ignored"
                function(gluster_event)
                return "OK"
```

Example 2

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

```
def __call__(self, form, field):
        if current_app.testing:
            return True

        if request.json:
            challenge = request.json.get('recaptcha_challenge_field', '')
            response = request.json.get('recaptcha_response_field', '')
        else:
            challenge = request.form.get('recaptcha_challenge_field', '')
            response = request.form.get('recaptcha_response_field', '')
        remote_ip = request.remote_addr

        if not challenge or not response:
            raise ValidationError(field.gettext(self.message))

        if not self._validate_recaptcha(challenge, response, remote_ip):
            field.recaptcha_error = 'incorrect-captcha-sol'
            raise ValidationError(field.gettext(self.message))
```

Example 3

Project: *comport* Author: *codeforamerica* File: *views.py* BSD 3-Clause "New" or "Revised" License 6 vo

```
def use_of_force():
    username = request.authorization.username
    extractor = Extractor.query.filter_by(username=username).first()
    department = extractor.first_department()
    request_json = request.json
    added_rows = 0
    updated_rows = 0

    uof_class = getattr(importlib.import_module("comport.data.models"), "UseOfForc

    for incident in request_json['data']:
        added = uof_class.add_or_update_incident(department, incident)
        if added is True:
            added_rows += 1
        elif added is False:
            updated_rows += 1

    extractor.next_month = None
    extractor.next_year = None
    extractor.save()
    return json.dumps({"added": added_rows, "updated": updated_rows})
```

Example 4

Project: *comport* Author: *codeforamerica* File: *views.py* BSD 3-Clause "New" or "Revised" License 6 vo

```
def officer_involved_shooting():
    username = request.authorization.username
    extractor = Extractor.query.filter_by(username=username).first()
    department = extractor.first_department()
    request_json = request.json
    added_rows = 0
    updated_rows = 0

    ois_class = getattr(importlib.import_module("comport.data.models"), "OfficerIr

    for incident in request_json['data']:
        added = ois_class.add_or_update_incident(department, incident)
        if added is True:
            added_rows += 1
        elif added is False:
            updated_rows += 1

    extractor.next_month = None
    extractor.next_year = None
    extractor.save()
    return json.dumps({"added": added_rows, "updated": updated_rows})
```

Example 5

Project: *comport* Author: *codeforamerica* File: *views.py* BSD 3-Clause "New" or "Revised" License 6 vo

```
def complaints():
    username = request.authorization.username
    extractor = Extractor.query.filter_by(username=username).first()
    department = extractor.first_department()
    request_json = request.json
    added_rows = 0
    updated_rows = 0

    complaint_class = getattr(importlib.import_module("comport.data.models"), "Cit
```

```
    for incident in request_json['data']:
        added = complaint_class.add_or_update_incident(department, incident)
        if added is True:
            added_rows += 1
        elif added is False:
            updated_rows += 1

    extractor.next_month = None
    extractor.next_year = None
    extractor.save()
    return json.dumps({"added": added_rows, "updated": updated_rows})
```

Example 6

Project: *comport* Author: *codeforamerica* File: *views.py* BSD 3-Clause "New" or "Revised" License 6 vo

```
def pursuits():
    username = request.authorization.username
    extractor = Extractor.query.filter_by(username=username).first()
    department = extractor.first_department()
    request_json = request.json
    added_rows = 0
    updated_rows = 0

    pursuit_class = getattr(importlib.import_module("comport.data.models"), "Pursu

    for incident in request_json['data']:
        added = pursuit_class.add_or_update_incident(department, incident)
        if added is True:
            added_rows += 1
        elif added is False:
            updated_rows += 1

    extractor.next_month = None
    extractor.next_year = None
    extractor.save()
    return json.dumps({"added": added_rows, "updated": updated_rows})
```

Example 7

Project: *iris* Author: *doitintl* File: *main.py* MIT License 6 vo

```
def tag_one():
    data = json.loads(base64.b64decode(request.json['message']['data']))
    logging.info(data)
    try:
        method_name = data['protoPayload']['methodName']
        for plugin in Plugin.plugins:
            if plugin.is_on_demand():
                for method in plugin.methodsNames():
                    if method.lower() in method_name.lower():
                        gcp_object = plugin.get_gcp_object(data)
                        if gcp_object is not None:
                            project_id = data['resource']['labels'][
                                'project_id']
                            logging.info("Calling tag one for %s", plugin.__class_
                            plugin.tag_one(gcp_object, project_id)
                            plugin.do_batch()
    except Exception as e:
```

```
        logging.error(e)
    return 'ok', 200
```

Example 8

Project: *door-monitor* Author: *Chris-Johnston* File: *sensor.py* MIT License 6 vo

```
def log_sensor():
    """
    Logs sensor data.
    """
    with get_db() as DB:
        c = DB.cursor()
        cause = request.json["cause"]
        for sensor in request.json["sensors"]:
            name = sensor["name"]
            state_bool = sensor["state"]
            state = 1 if state_bool else 0

            values = (name, state, cause)
            c.execute("INSERT INTO log (name, state, cause) VALUES (?, ?, ?)", val

            # TODO map the cause to a string
            notify_webhooks((name, state_bool, cause, state_mapping[state_bool], c
        DB.commit()

    return "Ok"
```

Example 9

Project: *botbuilder-python* Author: *microsoft* File: *app.py* MIT License 6 vo

```
def messages():
    # Main bot message handler.
    if "application/json" in request.headers["Content-Type"]:
        body = request.json
    else:
        return Response(status=415)

    activity = Activity().deserialize(body)
    auth_header = (
        request.headers["Authorization"] if "Authorization" in request.headers els
    )

    async def aux_func(turn_context):
        await BOT.on_turn(turn_context)

    try:
        task = LOOP.create_task(
            ADAPTER.process_activity(activity, auth_header, aux_func)
        )
        LOOP.run_until_complete(task)
        return Response(status=201)
    except Exception as exception:
        raise exception
```

Example 10

Project: *botbuilder-python* Author: *microsoft* File: *app.py* MIT License 6 vo

```
def messages():
    # Main bot message handler.
    if "application/json" in request.headers["Content-Type"]:
        body = request.json
    else:
        return Response(status=415)

    activity = Activity().deserialize(body)
    auth_header = (
        request.headers["Authorization"] if "Authorization" in request.headers els
    )

    try:
        task = LOOP.create_task(
            ADAPTER.process_activity(activity, auth_header, BOT.on_turn)
        )
        LOOP.run_until_complete(task)
        return Response(status=201)
    except Exception as exception:
        raise exception
```

Example 11

Project: *botbuilder-python* Author: *microsoft* File: *app.py* MIT License 6 vo

```
def messages():
    # Main bot message handler.
    if "application/json" in request.headers["Content-Type"]:
        body = request.json
    else:
        return Response(status=415)

    activity = Activity().deserialize(body)
    auth_header = (
        request.headers["Authorization"] if "Authorization" in request.headers els
    )

    try:
        print("about to create task")
        print("about to run until complete")
        run_coroutine(ADAPTER.process_activity(activity, auth_header, BOT.on_turn)
        print("is now complete")
        return Response(status=201)
    except Exception as exception:
        raise exception
```

Example 12

Project: *botbuilder-python* Author: *microsoft* File: *app.py* MIT License 6 vo

```
def messages():
    # Main bot message handler.
    if "application/json" in request.headers["Content-Type"]:
        body = request.json
    else:
        return Response(status=415)

    activity = Activity().deserialize(body)
    auth_header = (
        request.headers["Authorization"] if "Authorization" in request.headers els
    )
```

```
    try:
        task = LOOP.create_task(
            ADAPTER.process_activity(activity, auth_header, BOT.on_turn)
        )
        LOOP.run_until_complete(task)
        return Response(status=201)
    except Exception as exception:
        raise exception
```

Example 13

Project: *botbuilder-python* Author: *microsoft* File: *app.py* MIT License 6 vo

```
def messages():
    # Main bot message handler.
    if "application/json" in request.headers["Content-Type"]:
        body = request.json
    else:
        return Response(status=415)

    activity = Activity().deserialize(body)
    auth_header = (
        request.headers["Authorization"] if "Authorization" in request.headers els
    )

    try:
        task = LOOP.create_task(
            ADAPTER.process_activity(activity, auth_header, BOT.on_turn)
        )
        LOOP.run_until_complete(task)
        return Response(status=201)
    except Exception as exception:
        raise exception
```

Example 14

Project: *botbuilder-python* Author: *microsoft* File: *main.py* MIT License 6 vo

```
def messages():
    """Main bot message handler."""
    if "application/json" in request.headers["Content-Type"]:
        body = request.json
    else:
        return Response(status=415)

    activity = Activity().deserialize(body)
    auth_header = (
        request.headers["Authorization"] if "Authorization" in request.headers els
    )

    async def aux_func(turn_context):
        await BOT.on_turn(turn_context)

    try:
        task = LOOP.create_task(
            ADAPTER.process_activity(activity, auth_header, aux_func)
        )
        LOOP.run_until_complete(task)
        return Response(status=201)
```

```
    except Exception as exception:
        raise exception
```

Example 15

Project: *har-sanitizer* Author: *google* File: *harsan_api.py* Apache License 2.0 6 vo

```
def get_wordlist():
  """Returns default HarSanitizer wordlist."""
  hs = HarSanitizer()

  try:
    if WORDLIST_PATH[:4] == "http":
      wordlist_json = json.loads(urllib2.urlopen(WORDLIST_PATH).read())
      wordlist = hs.load_wordlist(wordlist=wordlist_json)
    else:
      wordlist = hs.load_wordlist(wordlist_path=WORDLIST_PATH)
  except Exception:
    message = {"message": "Error: {} not found.".format(WORDLIST_PATH)}
    data = json.dumps(message, default=json_serial)
    return Response(data, 500, mimetype="application/json")

  data = json.dumps(wordlist, default=json_serial)
  return Response(data, 200, mimetype="application/json")
```

Example 16

Project: *har-sanitizer* Author: *google* File: *harsan_api.py* Apache License 2.0 6 vo

```
def get_mimetype_scrublist():
  """Returns default HarSanitizer mimeTypes scrub list."""
  hs = HarSanitizer()

  try:
    if MIMETYPES_PATH[:4] == "http":
      mimetype_scrub_list = json.loads(urllib2.urlopen(MIMETYPES_PATH).read())
    else:
      with open(MIMETYPES_PATH, "r") as mimetypes_file:
        mimetype_scrub_list = json.load(mimetypes_file)
  except Exception:
    message = {"message": "Error: {} not found.".format(MIMETYPES_PATH)}
    data = json.dumps(message, default=json_serial)
    return Response(data, 500, mimetype="application/json")

  data = json.dumps(mimetype_scrub_list, default=json_serial)
  return Response(data, 200, mimetype="application/json")
```

Example 17

Project: *jenca-authentication* Author: *bimlauncher* File: *authentication.py* MIT License 6 vo

```
def load_user_from_id(user_id):
    """
    Flask-Login ``user_loader`` callback.

    The ``user_id`` was stored in the session environment by Flask-Login.
    user_loader stores the returned ``User`` object in ``current_user`` during
    every flask request.

    See https://flask-login.readthedocs.org/en/latest/#flask.ext.login.LoginManage
```

```
    :param user_id: The ID of the user Flask is trying to load.
    :type user_id: string
    :return: The user which has the email address ``user_id`` or ``None`` if
        there is no such user.
    :rtype: ``User`` or ``None``.
    """
    url = urljoin(STORAGE_URL, 'users/{email}').format(email=user_id)
    response = requests.get(url, headers={'Content-Type': 'application/json'})

    if response.status_code == codes.OK:
        details = json.loads(response.text)
        return User(
            email=details['email'],
            password_hash=details['password_hash'],
        )
```

Example 18

Project: *jenca-authentication* Author: *bimlauncher* File: *authentication.py* MIT License 6 vo

```
def load_user_from_token(auth_token):
    """
    Flask-Login token-loader callback.

    See https://flask-login.readthedocs.org/en/latest/#flask.ext.login.LoginManage

    :param auth_token: The authentication token of the user Flask is trying to
        load.
    :type user_id: string
    :return: The user which has the given authentication token or ``None`` if
        there is no such user.
    :rtype: ``User`` or ``None``.
    """
    response = requests.get(
        urljoin(STORAGE_URL, '/users'),
        headers={'Content-Type': 'application/json'},
    )

    for details in json.loads(response.text):
        user = User(
            email=details['email'],
            password_hash=details['password_hash'],
        )
        if user.get_auth_token() == auth_token:
            return user
```

Example 19

Project: *jenca-authentication* Author: *bimlauncher* File: *storage.py* MIT License 6 vo

```
def create_user():
    """
    Create a new user. See ``users_route`` for details.
    """
    email = request.json['email']
    password_hash = request.json['password_hash']

    if load_user_from_id(email) is not None:
        return jsonify(
            title='There is already a user with the given email address.',
```

```
            detail='A user already exists with the email "{email}"'.format(
                email=email),
        ), codes.CONFLICT

    user = User(email=email, password_hash=password_hash)
    db.session.add(user)
    db.session.commit()

    return jsonify(email=email, password_hash=password_hash), codes.CREATED
```

Example 20

Project: *delay-queue* Author: *lgphone* File: *manager.py* GNU General Public License v3.0 6 vo

```
def task():
    if request.method == 'GET':
        task_id = request.values.get('task_id')
        if task_id:
            data = task_manager.list_task(task_id)
        else:
            data = task_manager.list_task()

        return jsonify({'status': 100, 'data': data})

    if request.method == 'POST':
        # 有delay 参数设置为延迟10秒执行
        if request.json and request.json.get('delay'):
            result = wait_cal.delay(692, 28, delay=int(time.time() * 1000) + 10000
        else:
            result = wait_cal.delay(692, 28)
        return jsonify({'status': 100, 'data': result})

    if request.method == 'DELETE':
        task_id = request.values.get('task_id')
        if task_id:
            task_manager.delete_task(task_id)
        return jsonify({'status': 100, 'data': None})
```

Example 21

Project: *tamil-news-classification* Author: *vanangamudi* File: *nlp_template.py* GNU General Public License v3.0 6 vo

```
def _predict():
            print(' requests incoming..')
            sentence = []
            try:
                input_string = word_tokenize(request.json["text"].lower())
                sentence.append([VOCAB[w] for w in input_string] + [VOCAB['EOS']])
                dummy_label = LongVar([0])
                sentence = LongVar(sentence)
                input_ = [0], (sentence,), (0, )
                output, attn = model(input_)
                #print(LABELS[output.max(1)[1]], attn)
                nwords = len(input_string)
                return jsonify({
                    "result": {
                        'sentence': input_string,
                        'attn': ['{:0.4f}'.format(i) for i in attn.squeeze().data.c
                        'probs': ['{:0.4f}'.format(i) for i in output.exp().squeeze
                        'label': LABELS[output.max(1)[1].squeeze().data.cpu().numpy
```

```
                }
            })

        except Exception as e:
            print(e)
            return jsonify({"result":"model failed"})
```

Example 22

Project: *orcid-service* Author: *adsabs* File: *views.py* MIT License 6 vo

```
def orcid_profile(orcid_id):
    '''Get/Set /[orcid-id]/orcid-profile - all communication exclusively in JSON''
    payload, headers = check_request(request)
    if request.method == 'GET':
        r = current_app.client.get(current_app.config['ORCID_API_ENDPOINT'] + '/'
                           headers=headers)
    else:
        r = current_app.client.post(current_app.config['ORCID_API_ENDPOINT'] + '/'
                           json=payload, headers=headers)

    # save the profile data (just in case the user revokes access_token, we can st
    # from our local data); however - normally the updater should grab the latest
    if r.status_code == 200:
        update_profile(orcid_id, r.text)

    return r.text, r.status_code
```

Example 23

Project: *gitlab-freak* Author: *Pegase745* File: *__init__.py* MIT License 5 vo

```
def dispatcher():
    """Dispatcher for Gitlab webhook triggering."""
    data = request.json
    kind = data.get('object_kind')
    content = data.get('object_attributes')

    if (kind in 'issue') and (content.get('action') in 'open'):
        # when opening an issue, create a Trello card and comment on Gitlab
        try:
            link = ProjectHasBoard.by_project(content.get('project_id'))
        except Exception, e:
            # in case of orm_exc.NoResultFound
            app.logger.error(e)

        opening_list = trello.boards.get_list(link.board_id)[0]
        app.logger.info('Creating a card for issue #%s on %s list.' %
                        (content.get('iid'), opening_list.get('name')))
        card = trello.cards.new(
            '#%s %s' % (
                content.get('iid'), content.get('title')),
            opening_list.get('id'),
            '%s \n\n %s' % (content.get('description'), content.get('url')))

        # create link between a card and an issue
        ilink, created = get_or_create(
            db.session, IssueHasCard,
            issue_id=content.get('iid'), card_id=card.get('id'))

        if created:
```

```
            # create a comment in gitlab with card's shortUrl
            git.createissuewallnote(
                content.get('project_id'), content.get('iid'),
                'Created Trello card -> %s' % card.get('shortUrl'))
```

Example 24

Project: *gitlab-freak* Author: *Pegase745* File: *__init__.py* MIT License 5 vo

```
def register():
    """Register a project for dependency monitoring."""
    data = request.json
    project_id = data.get('project_id')
    project_type = data.get('project_type')

    # Fetch dependenies from repository
    dependencies = {
        'nodejs': nodeDepsFetcher,
    }

    try:
        deps = dependencies[project_type](project_id)
    except Exception, e:
        app.logger.error(e)
```

Example 25

Project: *gitlab-freak* Author: *Pegase745* File: *__init__.py* MIT License 5 vo

```
def unregister():
    """Unregister a project for dependency monitoring."""
    data = request.json
    project_id = data.get('project_id')

    try:
        delDeps = db.session.query(ProjectDependency)\
            .filter_by(project_id=project_id)\
            .delete(synchronize_session=False)
        db.session.commit()
    except Exception, e:
        app.logger.error(e)
        db.session.rollback()
```

Example 26

Project: *Mastering-Python-Networking-Second-Edition* Author: *PacktPublishing*
File: *chapter9_6.py* MIT License 5 vo

```
def new_device():
    device = Device()
    device.import_data(request.json)
    db.session.add(device)
    db.session.commit()
    return jsonify({}), 201, {'Location': device.get_url()}
```

Example 27

Project: *Mastering-Python-Networking-Second-Edition* Author: *PacktPublishing*
File: *chapter9_6.py* MIT License 5 vo

```
def edit_device(id):
    device = Device.query.get_or_404(id)
    device.import_data(request.json)
    db.session.add(device)
    db.session.commit()
    return jsonify({})
```

Example 28

Project: *Mastering-Python-Networking-Second-Edition* Author: *PacktPublishing*
File: *chapter9_7.py* MIT License 5 vo

```
def edit_device(id):
    device = Device.query.get_or_404(id)
    device.import_data(request.json)
    db.session.add(device)
    db.session.commit()
    return jsonify({})
```

Example 29

Project: *ponygifbot* Author: *Katharine* File: *ponygifbot.py* MIT License 5 vo

```
def handle_update():
    if 'inline_query' not in request.json:
        return 'u wot mate?'
    query = request.json['inline_query']
    gevent.spawn(ponies.handle_request, query['id'], query['query'])
    return ''
```

Example 30

Project: *terraformize* Author: *naorlivne* File: *terraformize_endpoint.py* GNU Lesser General Public License v3.0 5 vo

```
def apply_terraform(module_path: str, workspace_name: str) -> Tuple[str, int]:
    """
    A REST endpoint to apply terraform modules at a given module path inside the n
    workspace

    Arguments:
        :param module_path:  the name of the subdirectory for the module inside th
        "terraform apply" at
        :param workspace_name: the name of the workspace to run "terraform apply"

    Returns:
        :return return_body: a JSON of the stdout & stderr from the terraform run
        :return terraform_return_code: the terraform return code

    Exceptions:
        :except FileNotFoundError: will return HTTP 404 with a JSON of the stderr
        "terraform apply"
    """
    try:
        terraform_object = Terraformize(workspace_name, configuration["terraform_n
                                        terraform_bin_path=configuration["terrafor
        terraform_return_code, terraform_stdout, terraform_stderr = terraform_obje
            request.json, configuration["parallelism"]
        )
        return_body = jsonify({
```

```
            "init_stdout": terraform_object.init_stdout,
            "init_stderr": terraform_object.init_stderr,
            "stdout": terraform_stdout,
            "stderr": terraform_stderr
        })
        terraform_return_code = terraform_return_code_to_http_code(int(terraform_r
        return return_body, terraform_return_code
    except FileNotFoundError as error_log:
        return jsonify({"error": str(error_log)}), 404
```

Example 31

Project: *terraformize* Author: *naorlivne* File: *terraformize_endpoint.py* GNU Lesser General Public License v3.0 5 vo

```
def destroy_terraform(module_path: str, workspace_name: str) -> Tuple[str, int]:
    """
    A REST endpoint to destroy terraform modules at a given module path inside the
    workspace

    Arguments:
        :param module_path:  the name of the subdirectory for the module inside th
        "terraform destroy" at
        :param workspace_name: the name of the workspace to run "terraform destroy

    Returns:
        :return return_body: a JSON of the stdout & stderr from the terraform run
        :return terraform_return_code: the terraform return code

    Exceptions:
        :except FileNotFoundError: will return HTTP 404 with a JSON of the stderr
        "terraform destroy"
    """
    try:
        terraform_object = Terraformize(workspace_name, configuration["terraform_n
                                        terraform_bin_path=configuration["terrafor
        terraform_return_code, terraform_stdout, terraform_stderr = terraform_obje
            request.json, configuration["parallelism"]
        )
        return_body = jsonify({
            "init_stdout": terraform_object.init_stdout,
            "init_stderr": terraform_object.init_stderr,
            "stdout": terraform_stdout,
            "stderr": terraform_stderr
        })
        return return_body, terraform_return_code_to_http_code(int(terraform_retur
    except FileNotFoundError as error_log:
        return jsonify({"error": str(error_log)}), 404
```

Example 32

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

```
def __init__(self, formdata=_Auto, obj=None, prefix='', csrf_context=None,
                 secret_key=None, csrf_enabled=None, *args, **kwargs):

        if csrf_enabled is None:
            csrf_enabled = current_app.config.get('WTF_CSRF_ENABLED', True)

        self.csrf_enabled = csrf_enabled
```

```
        if formdata is _Auto:
            if self.is_submitted():
                formdata = request.form
                if request.files:
                    formdata = formdata.copy()
                    formdata.update(request.files)
                elif request.json:
                    formdata = werkzeug.datastructures.MultiDict(request.json)
            else:
                formdata = None

        if self.csrf_enabled:
            if csrf_context is None:
                csrf_context = session
            if secret_key is None:
                # It wasn't passed in, check if the class has a SECRET_KEY
                secret_key = getattr(self, "SECRET_KEY", None)

            self.SECRET_KEY = secret_key
        else:
            csrf_context = {}
            self.SECRET_KEY = ''
        super(Form, self).__init__(formdata, obj, prefix,
                                   csrf_context=csrf_context,
                                   *args, **kwargs)
```

Example 33

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

```
def new_post_comment(id):
    post = Post.query.get_or_404(id)
    comment = Comment.from_json(request.json)
    comment.author = g.current_user
    comment.post = post
    db.session.add(comment)
    db.session.commit()
    return jsonify(comment.to_json()), 201, \
        {'Location': url_for('api.get_comment', id=comment.id,
                             _external=True)}
```

Example 34

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

```
def new_post():
    post = Post.from_json(request.json)
    post.author = g.current_user
    db.session.add(post)
    db.session.commit()
    return jsonify(post.to_json()), 201, \
        {'Location': url_for('api.get_post', id=post.id, _external=True)}
```

Example 35

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

```
def edit_post(id):
    post = Post.query.get_or_404(id)
    if g.current_user != post.author and \
```

```
            not g.current_user.can(Permission.ADMINISTER):
        return forbidden('Insufficient permissions')
    post.body = request.json.get('body', post.body)
    db.session.add(post)
    return jsonify(post.to_json())
```

Example 36

Project: *DJFeet* Author: *libre-man* File: *web.py* MIT License 5 vo

```
def set_config():
    if app.got_options:
        return jsonify(ok=False), 412
    app.got_options = True
    app.queue.put((OPTIONS, request.json))
    return jsonify(ok=True)
```

Example 37

Project: *DJFeet* Author: *libre-man* File: *web.py* MIT License 5 vo

```
def add_music():
    try:
        app.queue.put_nowait(
            (PROCESS_SONG, request.json['file_location'], request.json['id']))
        return jsonify(ok=True)
    except queue.Full:
        return jsonify(ok=False)
```

Example 38

Project: *DJFeet* Author: *libre-man* File: *web.py* MIT License 5 vo

```
def remove_music():
    try:
        app.queue.put_nowait(
            (DELETE_SONG, request.json['file_location'], request.json['id']))
        return jsonify(ok=True)
    except queue.Full:
        return jsonify(ok=False)
```

Example 39

Project: *DJFeet* Author: *libre-man* File: *web.py* MIT License 5 vo

```
def im_alive():
    requests.post(
        app.config['REMOTE'] + "/im_alive/",
        json={
            'id': app.config['ID'],
            'options': Config.get_all_options(),
        })
```

Example 40

Project: *PythonMicroservicesDevelopment_Code* Author: *mtianyan* File: *05_cache.py* Apache License 2.0 5 vo

```
def change_user(user_id):
        user = request.json
        # setting a new timestamp
        user['modified'] = _time2etag()
        _USERS[user_id] = user
        resp = jsonify(user)
        resp.set_etag(user['modified'])
        return resp
```

Example 41

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

```
def verify_token():
    key = current_app.config['pub_key']
    try:
        token = request.json['access_token']
        audience = request.json.get('audience', '')
        return jwt.decode(token, key, audience=audience)
    except Exception as e:
        return _400(str(e))
```

Example 42

Project: *comport* Author: *codeforamerica* File: *views.py* BSD 3-Clause "New" or "Revised" License 5 vo

```
def heartbeat():
    username = request.authorization.username
    extractor = Extractor.query.filter_by(username=username).first()

    # set the extractor last contact datetime to now
    now = datetime.now()
    extractor.last_contact = now
    extractor.save()

    # get the month and year to tell the extractor to start from
    next_month = extractor.next_month if extractor.next_month else now.month
    next_year = extractor.next_year if extractor.next_year else now.year

    #
    # build and send a Slack notification about this ping
    slack_body_lines = []
    extractor_department = extractor.first_department()
    if extractor_department:
        slack_body_lines.append('For: {}'.format(extractor_department.name))
    else:
        slack_body_lines.append('Username: {}'.format(username))

    slack_date_line = 'Replied with extraction start date: {}/{}'.format(next_mont

    slack_body_lines.append(slack_date_line)
    send_slack_message('Comport Pinged by Extractor!', slack_body_lines)

    #
    # remove records for this department from the incidents_updated table
    IncidentsUpdated.delete_records(department_id=extractor_department.id)

    # respond to the extractor
    heartbeat_response = json.dumps({"received": request.json, "nextMonth": next_
    return heartbeat_response
```

Example 43

Project: *PathDump* Author: *PathDump* File: *agent.py* Apache License 2.0 5 vo

```
def not_found (error):
    return make_response ( json.dumps ({'error': 'Not found'}), 404)
```

Example 44

Project: *PathDump* Author: *PathDump* File: *agent.py* Apache License 2.0 5 vo

```
def getpathdumppost():
    if not request.json or not 'api' in request.json:
        abort (404)
    else:
        content = pq.handlerequest (request.json, "pathdump")
        return content
```

Example 45

Project: *PathDump* Author: *PathDump* File: *agent.py* Apache License 2.0 5 vo

```
def getpathdumpget():
    if not request.json or not 'api' in request.json:
        abort(404)
    else:
        content = pq.handlerequest (request.json, "pathdump")
        return content
```

Example 46

Project: *PathDump* Author: *PathDump* File: *agent.py* Apache License 2.0 5 vo

```
def getpathdumppost():
    if not request.json or not 'api' in request.json:
        abort (404)
    else:
        output = handleRequest (request.json)
        return json.dumps (output, default=json_util.default)
```

Example 47

Project: *PathDump* Author: *PathDump* File: *agent.py* Apache License 2.0 5 vo

```
def getpathdumpget():
    if not request.json or not 'api' in request.json:
        abort (404)
    else:
        output = handleRequest (request.json)
        return json.dumps (output, default=json_util.default)
```

Example 48

Project: *cisco-dnac-platform-webex-notifications* Author: *robertcsapo* File: *run.py* MIT License 5 vo

```
def sample():
    jsonFile = "outputdata.json"
    with open(jsonFile) as f:
```

```
        data = json.load(f)

    issueTitle = (data["details"]["Type"] + " " + data["details"]["Device"])
    issuePriority = data["details"]["Assurance Issue Priority"]
    issueSeverity = data["severity"]
    issueSummary = data["details"]["Assurance Issue Details"]

    data = "Warning Severity %s (%s)! %s - %s" % (issueSeverity, issuePriority, is
    webex(str(data))
    return("Sample data from -> %s" % jsonFile)
```

Example 49

Project: *cisco-dnac-platform-webex-notifications* Author: *robertcsapo* File: *run.py* MIT License 5 vo

```
def postSample():
    data = request.json

    issueTitle = (data["details"]["Type"] + " " + data["details"]["Device"])
    issuePriority = data["details"]["Assurance Issue Priority"]
    issueSeverity = data["severity"]
    issueSummary = data["details"]["Assurance Issue Details"]

    data = "Warning Severity %s (%s)! %s - %s" % (issueSeverity, issuePriority, is

    webex(str(data))
    return("Sample JSON Payload received")
```

Example 50

Project: *cisco-dnac-platform-webex-notifications* Author: *robertcsapo* File: *run.py* MIT License 5 vo

```
def dnacPayload():
    data = request.json
    if not len(data) == 0:

        issueTitle = (data["details"]["Type"] + " " + data["details"]["Device"])
        issuePriority = data["details"]["Assurance Issue Priority"]
        issueSeverity = data["severity"]
        issueSummary = data["details"]["Assurance Issue Details"]

        data = "Warning Severity %s (%s)! %s - %s" % (issueSeverity, issuePriority

        webex(str(data))
        return("Cisco DNA Center JSON Payload received")
    else:
        return("Connection Alive")
```