Python ruamel.yaml.YAML Examples

The following are 30 code examples for showing how to use *ruamel.yaml.YAML()*. These examples are extracted from open source projects. You can vote up the ones you like or vote down the ones you don't like, and go to the original project or source file by following the links above each example.

You may check out the related API usage on the sidebar.

You may also want to check out all available functions/classes of the module ruamel.yaml , or try the search function .

Example 1

Project: *maubot* Author: *maubot* File: zip.py License: GNU Affero General Public License v3.0 6 votes

```
def _read_meta(source) -> Tuple[ZipFile, PluginMeta]:
        try:
            file = ZipFile(source)
            data = file.read("maubot.yaml")
        except FileNotFoundError as e:
            raise MaubotZipMetaError("Maubot plugin not found") from e
        except BadZipFile as e:
            raise MaubotZipMetaError("File is not a maubot plugin") from e
        except KeyError as e:
            raise MaubotZipMetaError("File does not contain a maubot plugin definition") from e
        try:
            meta_dict = yaml.load(data)
        except (YAMLError, KeyError, IndexError, ValueError) as e:
            raise MaubotZipMetaError("Maubot plugin definition file is not valid YAML") from e
        try:
            meta = PluginMeta.deserialize(meta_dict)
        except SerializerError as e:
            raise MaubotZipMetaError("Maubot plugin definition in file is invalid") from e
        return file, meta
```

Example 2

Project: *maubot* Author: *maubot* File: build.py License: GNU Affero General Public License v3.0 6 votes

```
def read_meta(path: str) -> Optional[PluginMeta]:
    try:
        with open(os.path.join(path, "maubot.yaml")) as meta_file:
            try:
                meta_dict = yaml.load(meta_file)
            except YAMLError as e:
                print(Fore.RED + "Failed to build plugin: Metadata file is not YAML")
                print(Fore.RED + str(e) + Fore.RESET)
                return None
    except FileNotFoundError:
        print(Fore.RED + "Failed to build plugin: Metadata file not found" + Fore.RESET)
        return None
    try:
```

```
        meta = PluginMeta.deserialize(meta_dict)
    except SerializerError as e:
        print(Fore.RED + "Failed to build plugin: Metadata file is not valid")
        print(Fore.RED + str(e) + Fore.RESET)
        return None
    return meta
```

Example 3

Project: *calm-dsl* Author: *nutanix* File: entity.py License: Apache License 2.0 6 votes

```
def yaml_dump(cls, stream=sys.stdout):
        class MyRepresenter(SafeRepresenter):
            def ignore_aliases(self, data):
                return True

        yaml = YAML(typ="safe")
        yaml.default_flow_style = False
        yaml.Representer = MyRepresenter

        types = EntityTypeBase.get_entity_types()

        for _, t in types.items():
            yaml.register_class(t)

        yaml.indent(mapping=2, sequence=4, offset=2)
        yaml.dump(cls, stream=stream)
```

Example 4

Project: *APIFuzzer* Author: *KissPeter* File: fuzz_utils.py License: GNU General Public License v3.0 6 votes

```
def get_api_definition_from_file(src_file):
    try:
        with open(src_file, mode='rb') as f:
            api_definition = f.read()
        try:
            return json.loads(api_definition.decode('utf-8'))
        except ValueError as e:
            print('Failed to load input as JSON, maybe YAML?')
        try:
            yaml = YAML(typ='safe')
            return yaml.load(api_definition)
        except (TypeError, ScannerError) as e:
            print('Failed to load input as YAML:{}'.format(e))
            raise e
    except (Exception, FileNotFoundError):
        print('Failed to parse input file, exit')
        raise FailedToParseFileException
```

Example 5

Project: *xbbg* Author: *alpha-xone* File: param.py License: Apache License 2.0 6 votes

```
def to_hour(num) -> str:
    """
    Convert YAML input to hours
```

```
    Args:
        num: number in YMAL file, e.g., 900, 1700, etc.

    Returns:
        str

    Examples:
        >>> to_hour(900)
        '09:00'
        >>> to_hour(1700)
        '17:00'
    """
    to_str = str(int(num))
    return pd.Timestamp(f'{to_str[:-2]}:{to_str[-2:]}').strftime('%H:%M')
```

Example 6

Project: *borgmatic* Author: *witten* File: generate.py License: GNU General Public License v3.0 6 votes

```
def write_configuration(config_filename, rendered_config, mode=0o600):
    '''
    Given a target config filename and rendered config YAML, write it out to file. Create any
    containing directories as needed.
    '''
    if os.path.exists(config_filename):
        raise FileExistsError('{} already exists. Aborting.'.format(config_filename))

    try:
        os.makedirs(os.path.dirname(config_filename), mode=0o700)
    except (FileExistsError, FileNotFoundError):
        pass

    with open(config_filename, 'w') as config_file:
        config_file.write(rendered_config)

    os.chmod(config_filename, mode)
```

Example 7

Project: *borgmatic* Author: *witten* File: generate.py License: GNU General Public License v3.0 6 votes

```
def generate_sample_configuration(source_filename, destination_filename, schema_filename):
    '''
    Given an optional source configuration filename, and a required destination configuration
    filename, and the path to a schema filename in pykwalify YAML schema format, write out a
    sample configuration file based on that schema. If a source filename is provided, merge the
    parsed contents of that configuration into the generated configuration.
    '''
    schema = yaml.round_trip_load(open(schema_filename))
    source_config = None

    if source_filename:
        source_config = load.load_configuration(source_filename)

    destination_config = merge_source_configuration_into_destination(
        _schema_to_sample_configuration(schema), source_config
```

```
    )

    write_configuration(
        destination_filename,
        _comment_out_optional_configuration(_render_configuration(destination_config)),
    )
```

Example 8

Project: *allennlp* Author: *allenai* File: build_docs_config.py License: Apache License 2.0 6 votes

```
def main():
    yaml = YAML()
    opts = parse_args()

    source_yaml = yaml.load(Path(opts.source_yaml))

    nav_entries = build_api_toc(Path(opts.api_docs_path), Path(opts.docs_root))

    # Add version to name.
    source_yaml["site_name"] = f"AllenNLP {opts.docs_version}"

    # Find the yaml sub-object corresponding to the API table of contents.
    site_nav = source_yaml["nav"]
    for nav_obj in site_nav:
        if API_TOC_KEY in nav_obj:
            break
    nav_obj[API_TOC_KEY] = nav_entries

    with open(opts.target_yaml, "w") as f:
        yaml.dump(source_yaml, f)

    print(f"{opts.target_yaml} created")
```

Example 9

Project: *DeTTECT* Author: *rabobank-cdc* File: generic.py License: GNU General Public License v3.0 6 votes

```
def fix_date_and_remove_null(yaml_file, date, input_type='ruamel'):
    """
    Remove the single quotes around the date key-value pair in the provided yaml_file and remove any 'null'
values
    :param yaml_file: ruamel.yaml instance or location of YAML file
    :param date: string date value (e.g. 2019-01-01)
    :param input_type: input type can be a ruamel.yaml instance or list
    :return: YAML file lines in a list
    """
    _yaml = init_yaml()
    if input_type == 'ruamel':
        # ruamel does not support output to a variable. Therefore we make use of StringIO.
        file = StringIO()
        _yaml.dump(yaml_file, file)
        file.seek(0)
        new_lines = file.readlines()
    elif input_type == 'list':
        new_lines = yaml_file
    elif input_type == 'file':
```

```
        new_lines = yaml_file.readlines()

    fixed_lines = [l.replace('\'' + str(date) + '\'', str(date)).replace('null', '')
                   if REGEX_YAML_DATE.match(l) else
                   l.replace('null', '') for l in new_lines]

    return fixed_lines
```

Example 10

Project: *DeTTECT* Author: *rabobank-cdc* File: generic.py License: GNU General Public License v3.0 6 votes

```
def set_yaml_dv_comments(yaml_object):
    """
    Set all comments in the detection or visibility  YAML object when the 'comment' key-value pair is
missing or is None.
    This gives the user the flexibility to have  YAML files with missing 'comment' key-value pairs.
    :param yaml_object: detection or visibility object
    :return: detection or visibility object for which empty comments are filled with an empty string
    """
    yaml_object['comment'] = yaml_object.get('comment', '')
    if yaml_object['comment'] is None:
        yaml_object['comment'] = ''
    if 'score_logbook' in yaml_object:
        for score_obj in yaml_object['score_logbook']:
            score_obj['comment'] = score_obj.get('comment', '')
            if score_obj['comment'] is None:
                score_obj['comment'] = ''

    return yaml_object
```

Example 11

Project: *DeTTECT* Author: *rabobank-cdc* File: generic.py License: GNU General Public License v3.0 6 votes

```
def get_platform_from_yaml(yaml_content):
    """
    Read the platform field from the  YAML file supporting both string and list values.
    :param yaml_content: the content of the  YAML file containing the platform field
    :return: the platform value
    """
    platform = yaml_content.get('platform', None)
    if platform is None:
        return []
    if isinstance(platform, str):
        platform = [platform]
    platform = [p.lower() for p in platform if p is not None]

    if platform == ['all']:
        platform = 'all'
    else:
        valid_platform_list = []
        for p in platform:
            if p in PLATFORMS.keys():
                valid_platform_list.append(PLATFORMS[p])
        platform = valid_platform_list
    return platform
```

Example 12

Project: *armi* Author: *terrapower* File: settingsIO.py License: Apache License 2.0 6 votes

```
def _readYaml(self, stream, handleInvalids=True):
        """
        Read settings from a  YAML stream.

        Notes
        -----
        This is intended to replace the XML stuff as we converge on consistent input formats.
        """
        from armi.physics.thermalHydraulics import const  # avoid circular import

        yaml = YAML()
        tree = yaml.load(stream)
        if "settings" not in tree:
            raise exceptions.InvalidSettingsFileError(
                self.inputPath,
                "Missing the `settings:` header required in  YAML settings",
            )
        if const.ORIFICE_SETTING_ZONE_MAP in tree:
            raise exceptions.InvalidSettingsFileError(
                self.inputPath, "Appears to be an orifice_settings file"
            )
        caseSettings = tree[Roots.CUSTOM]
        self.inputVersion = tree["metadata"][Roots.VERSION]
        for settingName, settingVal in caseSettings.items():
            self._applySettings(settingName, settingVal)
```

Example 13

Project: *armi* Author: *terrapower* File: geometry.py License: Apache License 2.0 6 votes

```
def _readYaml(self, stream):
        """
        Read geometry from yaml.

        Notes
        -----
        This is intended to replace the XML format as we converge on
        consistent inputs.
        """
        yaml = YAML()
        tree = yaml.load(stream)
        tree = INPUT_SCHEMA(tree)
        self.assemTypeByIndices.clear()
        for _systemName, system in tree[INP_SYSTEMS].items():
            # no need to check for valid since the schema handled that.
            self.geomType = system[INP_GEOM]
            self.symmetry = system[INP_SYMMETRY]
            if INP_DISCRETES in system:
                self._read_yaml_discretes(system)
            elif INP_LATTICE in system:
                self._read_yaml_lattice(system)
```

Example 14

Project: *paasta* Author: *Yelp* File: spark_tools.py License: Apache License 2.0 6 votes

```
def _load_aws_credentials_from_yaml(yaml_file_path) -> Tuple[str, str]:
    with open(yaml_file_path, "r") as yaml_file:
        try:
            credentials_yaml = YAML().load(yaml_file.read())
        except Exception as e:
            print(
                PaastaColors.red(
                    "Encountered %s when trying to parse AWS credentials yaml %s. "
                    "Suppressing further output to avoid leaking credentials."
                    % (type(e), yaml_file_path)
                )
            )
            sys.exit(1)

        return (
            credentials_yaml["aws_access_key_id"],
            credentials_yaml["aws_secret_access_key"],
        )
```

Example 15

Project: *nornir-workshop* Author: *dmfigol* File: utils.py License: MIT License 6 votes

```
def get_devices_conn_params() -> Dict[str, Dict[str, str]]:
    """Creates a dictionary of connection parameters for SSH"""
    result: Dict[str, Dict[str, str]] = {}
    yaml = YAML()
    with open(HOSTS_FILE, 'r') as f:
        hosts = yaml.load(f)
    for device, device_details in hosts["devices"]["routers"].items():
        device_params = {
            "host": device_details["host"],
            "username": DEVICE_USERNAME,
            "password": DEVICE_PASSWORD,
            "device_type": DEVICE_TYPE,
            "timeout": CONNECTION_TIMEOUT,
            "global_delay_factor": constants.NETMIKO_GLOBAL_DELAY_FACTOR,
        }
        result[device] = device_params
    return result
```

Example 16

Project: *rasa_nlu* Author: *weizhenzhao* File: __init__.py License: Apache License 2.0 6 votes

```
def replace_environment_variables() -> None:
    """Enable yaml loader to process the environment variables in the yaml."""
    import re
    import os

    # noinspection RegExpRedundantEscape
    env_var_pattern = re.compile(r"^(.*)\$\{(.*)\}(.*)$")
    yaml.add_implicit_resolver('!env_var', env_var_pattern)

    def env_var_constructor(loader, node):
```

```
        """Process environment variables found in the YAML."""
        value = loader.construct_scalar(node)
        prefix, env_var, postfix = env_var_pattern.match(value).groups()
        return prefix + os.environ[env_var] + postfix

    yaml.SafeConstructor.add_constructor(u'!env_var', env_var_constructor)
```

Example 17

Project: *BentoML* Author: *bentoml* File: config.py License: Apache License 2.0 6 votes

```
def __init__(self, bento_service=None, kind="BentoService"):
        self.kind = kind
        self._yaml = YAML()
        self._yaml.default_flow_style = False
        self.config = self._yaml.load(
            BENTOML_CONFIG_YAML_TEPMLATE.format(
                kind=self.kind,
                bentoml_version=get_bentoml_deploy_version(),
                created_at=str(datetime.utcnow()),
            )
        )

        if bento_service is not None:
            self.config["metadata"].update(
                {
                    "service_name": bento_service.name,
                    "service_version": bento_service.version,
                }
            )
            self.config["env"] = bento_service.env.to_dict()
            self.config['apis'] = _get_apis_list(bento_service)
            self.config['artifacts'] = _get_artifacts_list(bento_service)
```

Example 18

Project: *PowerGenome* Author: *gschivley* File: util.py License: MIT License 6 votes

```
def write_case_settings_file(settings, folder, file_name):
    """Write a finalized dictionary to YAML file.

    Parameters
    ----------
    settings : dict
        A dictionary with settings
    folder : Path-like
        A Path object representing the folder for a single case/scenario
    file_name : str
        Name of the file.
    """
    folder.mkdir(exist_ok=True, parents=True)
    path_out = folder / file_name

    # yaml = YAML(typ="unsafe")
    _settings = deepcopy(settings)
    # for key, value in _settings.items():
    #     if isinstance(value, Path):
```

```
    #         _settings[key] = str(value)
    # yaml.register_class(Path)
    # stream = file(path_out, 'w')
    with open(path_out, "w") as f:
        yaml.dump(_settings, f)
```

Example 19

Project: *MLOps* Author: *rsethur* File: devops_pipeline_generator.py License: MIT License 6 votes

```
def main():
    input_file_path, output_file_path = getRuntimeArgs()
    yaml_parser = YAML()

    #parse the cli command and the parameters
    cli_cmd, params = parseCLISpec(input_file_path)
    print(cli_cmd)
    print(params)

    # load yml base template
    with open(YML_TEMPLATE_PATH, 'r') as yml_template_stream:
        yml = yaml_parser.load(yml_template_stream)

    # Update the base yaml with new cli spec
    updateYMLWithCLISpec(yaml_parser, cli_cmd, params, yml)

    # write it out
    yaml_parser.dump(yml, sys.stdout)
    with open(output_file_path, 'w') as output_stream:
        yaml_parser.dump(yml, output_stream)
```

Example 20

Project: *NURBS-Python* Author: *orbingol* File: exchange.py License: MIT License 5 votes

```
def import_yaml(file_name, **kwargs):
    """ Imports curves and surfaces from files in YAML format.

    .. note::

        Requires `ruamel.yaml <https://pypi.org/project/ruamel.yaml/>`_ package.

    Use ``jinja2=True`` to activate Jinja2 template processing. Please refer to the documentation for
details.

    :param file_name: name of the input file
    :type file_name: str
    :return: a list of rational spline geometries
    :rtype: list
    :raises GeomdlException: an error occurred reading the file
    """
    def callback(data):
        yaml = YAML()
        return yaml.load(data)

    # Check if it is possible to import 'ruamel.yaml'
    try:
```

```
        from ruamel.yaml import YAML
    except ImportError:
        raise exch.GeomdlException("Please install 'ruamel.yaml' package to use YAML format: pip install
ruamel.yaml")

    # Get keyword arguments
    delta = kwargs.get('delta', -1.0)
    use_template = kwargs.get('jinja2', False)

    # Read file
    file_src = exch.read_file(file_name)

    # Import data
    return exch.import_dict_str(file_src=file_src, delta=delta, callback=callback, tmpl=use_template)
```

Example 21

Project: *NURBS-Python* Author: *orbingol* File: exchange.py License: MIT License 5 votes

```
def export_yaml(obj, file_name):
    """ Exports curves and surfaces in YAML format.

    .. note::

        Requires `ruamel.yaml <https://pypi.org/project/ruamel.yaml/>`_ package.

    YAML format is also used by the `geomdl command-line application <https://github.com/orbingol/geomdl-
cli>`_
    as a way to input shape data from the command line.

    :param obj: input geometry
    :type obj: abstract.SplineGeometry, multi.AbstractContainer
    :param file_name: name of the output file
    :type file_name: str
    :raises GeomdlException: an error occurred writing the file
    """
    def callback(data):
        # Ref: https://yaml.readthedocs.io/en/latest/example.html#output-of-dump-as-a-string
        stream = StringIO()
        yaml = YAML()
        yaml.dump(data, stream)
        return stream.getvalue()

    # Check if it is possible to import 'ruamel.yaml'
    try:
        from ruamel.yaml import YAML
    except ImportError:
        raise exch.GeomdlException("Please install 'ruamel.yaml' package to use YAML format: pip install
ruamel.yaml")

    # Export data
    exported_data = exch.export_dict_str(obj=obj, callback=callback)

    # Write to file
    return exch.write_file(file_name, exported_data)
```

Example 22

Project: *telemetry* Author: *jupyter* File: test_register_schema.py License: BSD 3-Clause "New" or "Revised" License

5 votes

```
def test_register_schema_file():
    """
    Register schema from a file
    """
    schema = {
        '$id': 'test/test',
        'version': 1,
        'properties': {
            'something': {
                'type': 'string'
            },
        },
    }

    el = EventLog()

    yaml = YAML(typ='safe')
    with tempfile.NamedTemporaryFile(mode='w') as f:
        yaml.dump(schema, f)
        f.flush()

        f.seek(0)

        el.register_schema_file(f.name)

    assert schema in el.schemas.values()
```

Example 23

Project: *telemetry* Author: *jupyter* File: eventlog.py License: BSD 3-Clause "New" or "Revised" License

5 votes

```
def register_schema_file(self, filename):
        """
        Convenience function for registering a JSON schema from a filepath

        Supports both JSON & YAML files.
        """
        # Just use YAML loader for everything, since all valid JSON is valid YAML
        with open(filename) as f:
            self.register_schema(yaml.load(f))
```

Example 24

Project: *xbbg* Author: *alpha-xone* File: param.py License: Apache License 2.0

5 votes

```
def _load_yaml_(file_name):
    """
    Load assets infomation from file

    Args:
        file_name: file name

    Returns:
```

```
        dict
    """
    if not os.path.exists(file_name): return dict()

    with open(file_name, 'r', encoding='utf-8') as fp:
        return  YAML().load(stream=fp)
```

Example 25

Project: *borgmatic* Author: *witten* File: generate.py License: GNU General Public License v3.0 5 votes

```
def _render_configuration(config):
    '''
    Given a config data structure of nested OrderedDicts, render the config as  YAML and return it.
    '''
    dumper = yaml.YAML()
    dumper.indent(mapping=INDENT, sequence=INDENT + SEQUENCE_INDENT, offset=INDENT)
    rendered = io.StringIO()
    dumper.dump(config, rendered)

    return rendered.getvalue()
```

Example 26

Project: *borgmatic* Author: *witten* File: generate.py License: GNU General Public License v3.0 5 votes

```
def remove_commented_out_sentinel(config, field_name):
    '''
    Given a configuration CommentedMap and a top-level field name in it, remove any "commented out"
    sentinel found at the end of its  YAML  comments. This prevents the given field name from getting
    commented out by downstream processing that consumes the sentinel.
    '''
    try:
        last_comment_value = config.ca.items[field_name][RUAMEL_YAML_COMMENTS_INDEX][-1].value
    except KeyError:
        return

    if last_comment_value == '# {}\n'.format(COMMENTED_OUT_SENTINEL):
        config.ca.items[field_name][RUAMEL_YAML_COMMENTS_INDEX].pop()
```

Example 27

Project: *network-programmability-stream* Author: *dmfigol* File: test_krk_stp.py License: MIT License 5 votes

```
def load_config(config_file: str) -> Dict[str, Any]:
    yaml = YAML(typ="safe")
    dir_path = Path(__file__).parent
    with open(dir_path / config_file) as f:
        return yaml.load(f)
```

Example 28

Project: *network-programmability-stream* Author: *dmfigol* File: test_krk_vlans.py License: MIT License 5 votes

```
def load_config(config_file: str) -> Dict[str, Any]:
    yaml = YAML(typ="safe")
    dir_path = Path(__file__).parent
    with open(dir_path / config_file) as f:
        return yaml.load(f)
```

Example 29

Project: *atomic-reactor* Author: *containerbuildsystem* File: operator.py License: BSD 3-Clause "New" or "Revised" License 5 votes

```
def __init__(self, data):
        """
        Initialize a NamedPullspec

        :param data: Dict-like object in JSON/YAML data
                     in which the name and image can be found
        """
        self.data = data
```

Example 30

Project: *bioconda-utils* Author: *bioconda* File: recipe.py License: MIT License 5 votes

```
def __init__(self, recipe_dir, recipe_folder):
        if not recipe_dir.startswith(recipe_folder):
            raise RuntimeError(f"'{recipe_dir}' not inside '{recipe_folder}'")


        #: path to folder containing recipes
        self.basedir = recipe_folder
        #: relative path to recipe dir from folder containing recipes
        self.reldir = recipe_dir[len(recipe_folder):].strip("/")

        # Filled in by render()
        #: Parsed recipe YAML
        self.meta: Dict[str, Any] = {}

        # These will be filled in by load_from_string()
        #: Lines of the raw recipe file
        self.meta_yaml: List[str] = []
        # Filled in by update filter
        self.version_data: Dict[str, Any] = {}
        #: Original recipe before modifications (updated by load_from_string)
        self.orig: Recipe = deepcopy(self)
        #: Whether the recipe was loaded from a branch (update in progress)
        self.on_branch: bool = False
        #: For passing data around
        self.data: Dict[str, Any] = {}

        # for conda_render() and conda_release()
        self._conda_meta = None
        self._conda_tempdir = None
```