```python
from bs4 import BeautifulSoup
import os
import pandas as pd
import re

from extractors.direct_downloads import extract_with_request, extract_with_headless_browser
from extractors.direct_downloads import format_extracted_data, write_extracted_data
from models.registry import Registry
from utils.helpers import clean_str, date_to_iso, hexadecimal_to_decimal


class NZL(Registry):
    def __init__(self, **kwargs):
        """
        Child container for extracting, transforming, and loading registry data for New Zealand

        :param kwargs: see models.registry for details
        """

        [illegible]

        # assign constant values for final registry
        [illegible]
        [illegible]
        [illegible]
        [illegible]

        # assign registry input data source and filename
        [illegible]
        [illegible]
        [illegible]

        # assign registry input data extraction method
        [illegible]
        # assign updated-date scraping method for registry input data
        [illegible]
        # assign registry input data reading method
        [illegible]
        # assign registry data wrangling method
        [illegible]

        # extract, transform, and load registry data
        [illegible]

    def extract_registry(self):
        """
        Extracts registry input data from remote source

        :return: effect - creates [DATA_DIR|TEST_DIR]/[REGISTRY_INPUT_DIR]/[self.country_code]/[self.registry_filename] file
        """

        # extract abstracted registry input data
        [illegible]
                             [illegible]

        # read abstracted registry input data
        [illegible]
        [illegible]

        [illegible]
            # build test set if indicated
            [illegible]

        # initialize locations of detailed registry input data to extract
        [illegible]

        # initialize storage for aggregated detailed registry input data
        [illegible]

        # extract detailed registry input data
        [illegible]
            [illegible]

        # write extracted registry input data
        [illegible]

        # format extracted registry input data
        [illegible]

        # read registry input data
        [illegible]

    def scrape_registry_updated(self):
        """
        Scrapes and formats updated-date for registry input data from remote source

        :return: effect - modifies self.updated
        """

        [illegible]
        [illegible]
        [illegible]
        [illegible]
        [illegible]

        # assign updated-date for registry input data
        [illegible]

    def read_registry(self, registry=None):
        """
        Reads registry input data

        :param (str) registry: stringified html representing extracted registry input data
        :return: effect - modifies self.registry
        """

        [illegible]
            # handle when registry input data has been extracted but not in current session
            [illegible]

        # read registry input data
        [illegible]
        # prep registry input data
        [illegible]
        [illegible]

        # assign registry input data
        [illegible]

    def wrangle_registry(self):
        """
        Transforms registry data

        :return: effect - modifies self.wrangled_records
        """

        # loop through raw registrations
        [illegible]
            # initialize storage for wrangled representation of registration in focus
            [illegible]
                [illegible]
                [illegible]
                [illegible]
                [illegible]
                [illegible]
                [illegible]
            [illegible]

            [illegible]

            [illegible]
            [illegible]

            # update list of transformed values to populate final registry
            [illegible]

    @staticmethod
    def __format_registry(registry):
        """
        Parses and merges registry input data across multiple html documents

        :param (bs4.BeautifulSoup) registry: concatenated html documents to process
        :return: structured and merged transformation of 'registry' argument
        """

        # assign registry input data keys parsed from html document
        [illegible]
        # initialize storage for registry input data values
        [illegible]

        # loop through html documents
        [illegible]
            # loop through key-value element pairs
            [illegible]
                [illegible]
                [illegible]
                # build registry input data
                [illegible]

        [illegible]

        [illegible]

    @staticmethod
    def __format_colname(colname):
        """
        Formats column name for namedtuple compatibility

        :param (str) colname: column name from registry input data to format
        :return: formatted copy of 'colname' argument suitable for namedtuple indexing
        """

        [illegible]
        [illegible]

        [illegible]

    @staticmethod
    def __get_icao(icao):
        """
        Parses and formats aircraft Mode-S Code

        :param (str) icao: Mode-S Code values to extract from
        :return: formatted extraction from 'icao' argument
        """

        [illegible]
        [illegible]
        [illegible]
        [illegible]

        [illegible]

    @staticmethod
    def __get_name_and_address(registrant):
        """
        Parses and formats registrant name and address

        :param (str) registrant: registrant information to extract values from
        :return: formatted registrant name; formatted registrant address
        """

        [illegible]

        [illegible]

        [illegible]
            [illegible]
        [illegible]
            [illegible]

        [illegible]
```