Project 1 - Part 3

PART 3 – FUNCTIONS AND CLASSES

1. Write the introduction of your scenario.
2. Write the explanation of the program created above.
3. Provide your conclusion.

I am a medical doctor. I want to create a Python project that helps classify lung diseases. Lung disease is an umbrella term for issues of the lungs and can be categorised into categories such as infection, non-infectious, and malignancy. Each category (child class) contains details of the parent class but may include their unique features.

The code begins by initiating the parent class `LungDisease,` which contains the name and umbrella category "Lung Disease." The `LungDisease` takes in only one parameter. The parent class is initiated by the `__init__` method, which gives the attribute to the name and "Lung Disease" category to itself. A method of `get_info()` prints out the disease information, i.e. the name and main category. An example of this is creating an instance of pneumonia, an infectious lung disease that is still ultimately a lung disease. Thus, if we create an instance of pneumonia by `pneumonia = LungDisease("Pneumonia")` and print out the `get_info()` function, we will be able to obtain the general category and name of the disease.

The subsequent codes are the child class to the `LungDisease` class, which are `InfectiousDisease, NonInfectiousDisease,` and `Malignancy.` The respective child classes contain features such as subcategories, severity, contagious status, staging, and metastasis. The child class also retains information about the parent class and passes it through the `super().__init__().` To obtain the information of instances, the

`get_info()`, similar to the parent class, is also available for the child class to streamline the usage of this class.

The first child class is the `InfectiousDisease` class. The class has multiple parameters such as `severity`, `pathogen_type`, and `is_contagious`. This class is first initiated with the `__init__` method and calls in the details from the parent class using the `super()__init__(name)`. There are a few set attributes, such as `self.pathogen_type` set to the `pathogen_type`, `self.is_contagious` set to the `is_contagious`, `self.sub_category` set to "Infectious Disease", and `self.severity` set to `severity`. It uses the same `get_info()` method as the parent class, which provides the details of the disease. It also has an extra `get_pathogen_type()` method, which returns only the disease name and pathogen type.

The `NonInfectiousDisease` class is a subclass of `LungDisease` that focuses on non-infectious conditions. It introduces two new parameters: `severity` and `is_chronic`. Like the `InfectiousDisease` class, the `__init__` method calls the parent class constructor using `super().__init__(name)` to initialize shared attributes. Additionally, it defines specific attributes: `is_chronic`, which indicates whether the condition is chronic or acute, and `severity`, which describes the seriousness of the disease. The `sub_category` is set to "Non-Infectious Disease." The `NonInfectiousDisease` class also overrides the parent's `get_info()` method to provide additional details specific to non-infectious conditions, including whether the disease is chronic. Furthermore, it introduces a new method called `get_chronic_condition()`, which determines and returns whether the condition is chronic.

The last class, the `Malignancy` class, is the third child of the `LungDisease` class and focuses on diseases related to lung cancer. This class introduces parameters such as `cancer_stage`, `metastasis`, and `severity`. As

with the other subclasses, the `__init__` method starts by invoking the parent class's constructor with `super().__init__(name)`. It then defines its unique attributes: `cancer_stage`, which specifies the stage of cancer progression; `metastasis`, a boolean value indicating whether the cancer has spread to other parts of the body; `severity`, describing the seriousness of the malignancy; and `sub_category`, set explicitly to "Malignancy". The `Malignancy` class overrides the `get_info()` method to provide detailed information about the disease, including the cancer stage and the presence of metastasis. It also introduces the `prognosis()` method, which estimates the disease prognosis based on the cancer stage and metastasis status. This method determines whether the prognosis is good or poor, considering the severity of the condition.

In conclusion, each of the subclasses builds upon the foundation laid by the `LungDisease` class by enhancing with more specific attributes and methods respective to each disease subcategory. This class can be used to categorise subcategories of Lung Diseases without sacrificing details of general lung diseases. This can be seen with examples provided for each parent and child class.