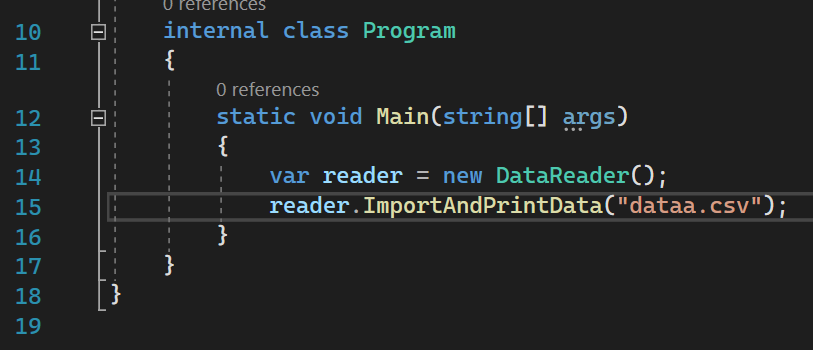
## Plik „Program.cs”

### Linia 15: Błąd w nazwie pliku

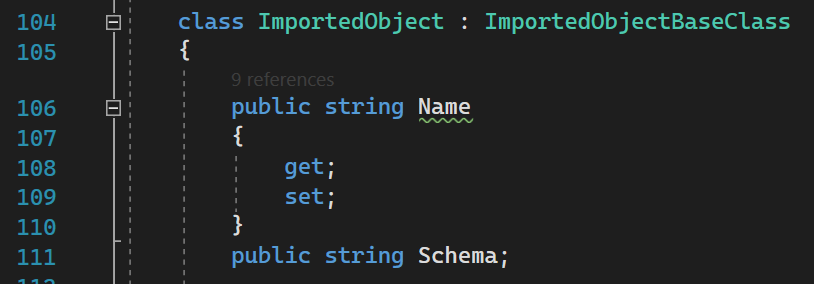
Należy zmienić nazwę pliku (*dataa.csv* => *data.csv*) w wywołaniu funkcji *DataReader.ImportAndPrintData*.



## Plik „DataReader.cs”

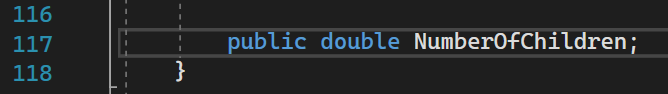
### Linie 106-110: Ponowna deklaracja właściwości „Name”

Właściwość „Name” została już zadeklarowana w klasie bazowej „ImportedObjectBaseClass”, wobec czego jej ponowna deklaracja w klasie „ImportedObject” może zostać usunięta.



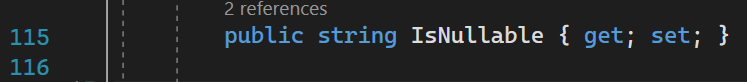
### Linia 117: Możliwość zmiany typu zmiennej „NumberOfChildren”

Ponieważ liczba potomków elementu zawsze będzie nieujemną liczbą całkowitą, typ zmiennej „NumberOfChildren” można zmienić z „double” na „uint”.



### Linia 115: Możliwość zmiany typu zmiennej „IsNullable”

Ponieważ właściwość „IsNullable” przechowuje jedynie informację typu prawda/fałsz, jej typ można zmienić ze „string” na „bool”. Powoduje to również potrzebę zmiany warunku w wyrażeniu warunkowym w linii 90 (*column.IsNullable == "1"* => *column.IsNullable*) oraz sposobu przypisywania wartości do pola podczas odczytu danych z pliku .csv). Inną możliwością byłaby zmiana typu na „char” – ułatwiłoby to pobieranie danych z pliku, jednak ponownie skomplikowało warunek wyrażenia warunkowego.



### Linie 27-40: Zmiana sposobu odwołania do elementu tablicy

Ponieważ w wierszu pliku .csv może być mniej niż 7 kolumn, bezpieczniejszym sposobem odwoływania się do odpowiednich elementów tablicy jest wykorzystanie metody *ElementAtOrDefault*.



### Linia 27: Błędny warunek pętli for

Znak nierówności w warunku pętli *for* musi zostać zmieniony z nieostrej na ostrą. Ponadto, ponieważ pierwszy wiersz pliku .csv zawiera jedynie schemat danych, może zostać pominięty przy wczytywaniu obiektów.



### Linia 16: Niepotrzebna inicjalizacja pierwszego elementu listy

Aby program działał poprawnie, listę należy zainicjalizować jako pustą.

