

PAKIET MATERIAŁÓW DYDAKTYCZNYCH

do kształcenia na odległość dla nauczycieli
informatyki klasy VII szkoły podstawowej

Materiał opracowany w ramach grantu przez Ośrodek Doskonalenia Nauczycieli
w Słupsku – **Jerzy Dorożko**

Projekt „Wsparcie placówek doskonalenia nauczycieli i bibliotek pedagogicznych w realizacji zadań związanych z przygotowaniem i wsparciem nauczycieli w prowadzeniu kształcenia na odległość”

Projekt współfinansowany ze środków Unii Europejskiej w ramach Europejskiego Funduszu Społecznego

SCENARIUSZ

SCENARIUSZ ZAJĘĆ DLA: uczniów klasy VII szkoły podstawowej

PROWADZONYCH PRZEZ: nauczyciela informatyki

TEMAT:

Żegnamy Scratcha – witamy Pythona!

WSTĘP:

W trakcie zajęć uczniowie poznają podstawowe różnice i podobieństwa występujące podczas programowania w środowisku programowania wizualnego Scratch, a środowiskiem tekstowym języka skryptowego Python. Założą konta w serwisie replit.com, poznają jego interfejs, a także dowiedzą się, czym jest zintegrowane środowisko programistyczne (IDE od ang. integrated development environment) i poznają podstawy korzystania z IDE dostępnego w replit.com.

CELE ZAJĘĆ – UCZEŃ POWINIEN:

- poznać podstawowe różnice między programowaniem w środowisku wizualnym a tekstowym,
- umieć poruszać się w interfejsie zintegrowanego środowiska programistycznego serwisu replit.com,
- potrafić napisać i uruchomić prosty program w języku skryptowym Python.

METODY PRACY:

- dyskusja,
- prezentacja,
- praca zespołowa,
- ćwiczenia praktyczne.

ŚRODKI DYDAKTYCZNE:

- komputery stacjonarne lub laptopy z dostępem do Internetu,
- bezpłatne konto użytkownika w serwisie <https://replit.com> [dostęp: 01.02.22],
- bezpłatne konto użytkownika w serwisie <https://scratch.mit.edu> [dostęp: 01.02.22].

PRZEWIDYWANY CZAS:

- 45 minut (1 godzina lekcyjna).

TREŚCI PROGRAMOWE – ZWIĄZEK Z PODSTAWĄ PROGRAMOWĄ:

- podstawa programowa kształcenia ogólnego dla szkół podstawowych – II etap edukacyjny – klasy IV-VIII, informatyka.

CELE KSZTAŁCENIA INFORMATYCZNEGO – WYMAGANIA OGÓLNE:

- I. Rozumienie, analizowanie i rozwiązywanie problemów na bazie logicznego i abstrakcyjnego myślenia, myślenia algorytmicznego i sposobów reprezentowania informacji.
- II. Programowanie i rozwiązywanie problemów z wykorzystaniem komputera oraz innych urządzeń cyfrowych: układanie i programowanie algorytmów, organizowanie, wyszukiwanie i udostępnianie informacji, posługiwanie się aplikacjami komputerowymi.
- III. Posługiwanie się komputerem, urządzeniami cyfrowymi i sieciami komputerowymi, w tym: znajomość zasad działania urządzeń cyfrowych i sieci komputerowych oraz wykonywania obliczeń i programów.
- IV. Rozwijanie kompetencji społecznych, takich jak: komunikacja i współpraca w grupie, w tym w środowiskach wirtualnych, udział w projektach zespołowych oraz organizacja i zarządzanie projektami.
- V. Przestrzeganie prawa i zasad bezpieczeństwa. Respektowanie prywatności informacji i ochrony danych, praw własności intelektualnej, etykiety w komunikacji i norm współżycia społecznego, ocena zagrożeń związanych z technologią i ich uwzględnienie dla bezpieczeństwa swojego i innych.

TREŚCI NAUCZANIA – WYMAGANIA SZCZEGÓŁOWE:

- I. Rozumienie, analizowanie i rozwiązywanie problemów. Uczeń:
 1. W algorytmicznym rozwiązywaniu problemu wyróżnia podstawowe kroki:
 - określenie problemu i celu do osiągnięcia,
 - analiza sytuacji problemowej,
 - opracowanie rozwiązania,

- sprawdzenie rozwiązania problemu dla przykładowych danych,
 - zapisanie rozwiązania w postaci schematu lub programu.
- II. Programowanie i rozwiązywanie problemów z wykorzystaniem komputera i innych urządzeń cyfrowych. Uczeń:
1. Gromadzi, porządkuje i selekcjonuje efekty swojej pracy oraz potrzebne zasoby w komputerze lub w innych urządzeniach oraz w środowiskach wirtualnych (w chmurze).
- III. Posługiwanie się komputerem, urządzeniami cyfrowymi i sieciami komputerowymi. Uczeń:
1. Wykorzystuje sieć komputerową (szkolną, sieć Internet):
 - a. do wyszukiwania potrzebnych informacji i zasobów edukacyjnych, nawigując między stronami,
 - b. jako medium komunikacyjne,
 - c. do pracy w wirtualnym środowisku (na platformie, w chmurze), stosując się do sposobów i zasad pracy w takim środowisku.
- IV. Rozwijanie kompetencji społecznych. Uczeń:
1. Uczestniczy w zespołowym rozwiązaniu problemu, posługując się technologią, taką jak:
 - poczta elektroniczna,
 - forum,
 - wirtualne środowisko kształcenia,
 - dedykowany portal edukacyjny.
- V. Przestrzeganie prawa i zasad bezpieczeństwa. Uczeń:
1. Uznaje i respektuje prawo do prywatności danych i informacji oraz prawo do własności intelektualnej.

TREŚCI NAUCZANIA – WYMAGANIA SZCZEGÓŁOWE:

- I. Programowanie i rozwiązywanie problemów z wykorzystaniem komputera i innych urządzeń cyfrowych. Uczeń:

1. Projektuje, tworzy i testuje programy w procesie rozwiązywania problemów.

W programach stosuje:

- instrukcje wejścia/wyjścia,
- wyrażenia arytmetyczne i logiczne,
- instrukcje warunkowe,
- instrukcje iteracyjne,
- funkcje,
- zmienne i tablice.

2. Projektuje, tworzy i testuje oprogramowanie sterujące robotem lub innym obiektem na ekranie lub w rzeczywistości.

3. Wyszukuje w sieci informacje potrzebne do realizacji wykonywanego zadania, stosując złożone postaci zapytań i korzysta z zaawansowanych możliwości wyszukiwarek.

II. Rozwijanie kompetencji społecznych. Uczeń:

1. Bierze udział w różnych formach współpracy, takich jak: programowanie w parach lub w zespole, realizacja projektów, uczestnictwo w zorganizowanej grupie uczących się. Projektuje, tworzy i prezentuje efekty wspólnej pracy.

III. Przestrzeganie prawa i zasad bezpieczeństwa. Uczeń:

1. Opisuje kwestie etyczne związane z wykorzystaniem komputerów i sieci komputerowych, takie jak:

- bezpieczeństwo,
- cyfrowa tożsamość,
- prywatność,
- własność intelektualna,
- równy dostęp do informacji,
- dzielenie się informacją.

PROPONOWANY PRZEBIEG ZAJĘĆ:

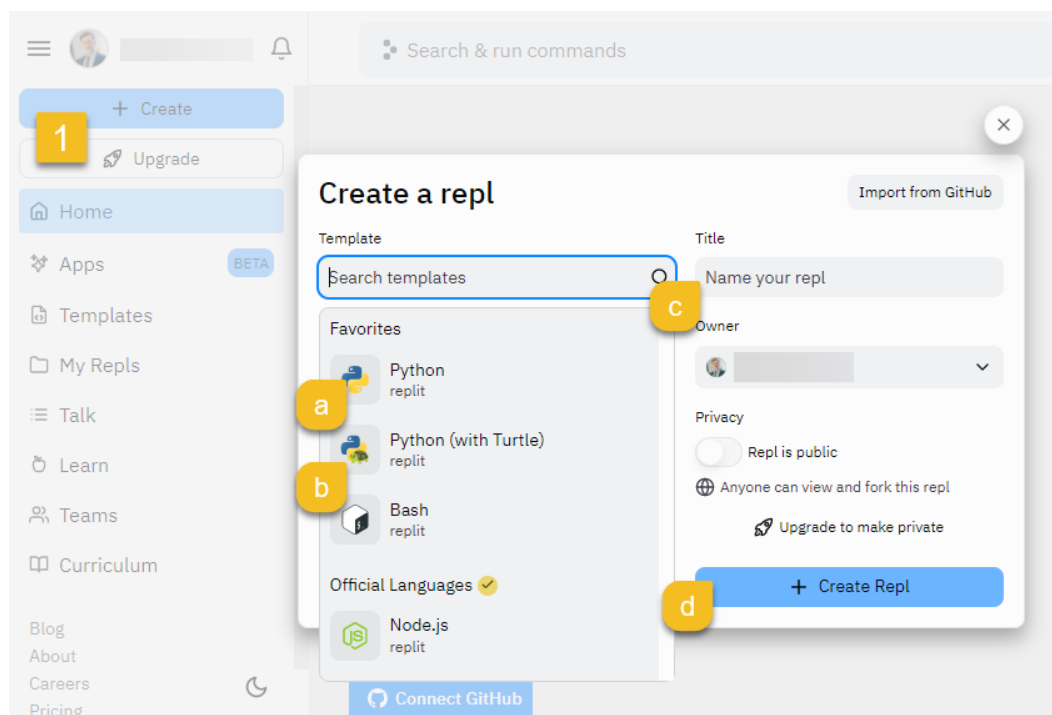
Wprowadzenie w tematykę i integracja grupy:

Nauczyciel przypomina podstawowe cechy środowiska programowania wizualnego Scratch oraz cel grupowania bloków poleceń w grupy (Ruch, Wygląd, Dźwięk itd.). Przełącza również język interfejsu na angielski, zwracając uwagę na jego powszechne stosowanie w świecie programowania (interfejsy programów, porozumiewanie się programistów w grupach międzynarodowych).

Uczniowie otrzymują również niezbędne informacje ukazujące różnorodność języków programowania oraz przyczynę ich mnogości. Dowiadują się również, iż kolejnym językiem, którego podstawy poznają, będzie Python. Korzystając z wyszukiwarki internetowej, poszukują przykładów jego wykorzystania we współczesnym świecie, po czym dzielą się zdobytymi informacjami.

Część zasadnicza:

Uczniowie pod kierunkiem nauczyciela zakładają bezpłatne konta pod adresem <https://replit.com/signup> [dostęp: 01.02.22]. Przypominają zasady bezpieczeństwa związane z udostępnianiem danych osobowych oraz tworzeniem i przechowywaniem haseł.

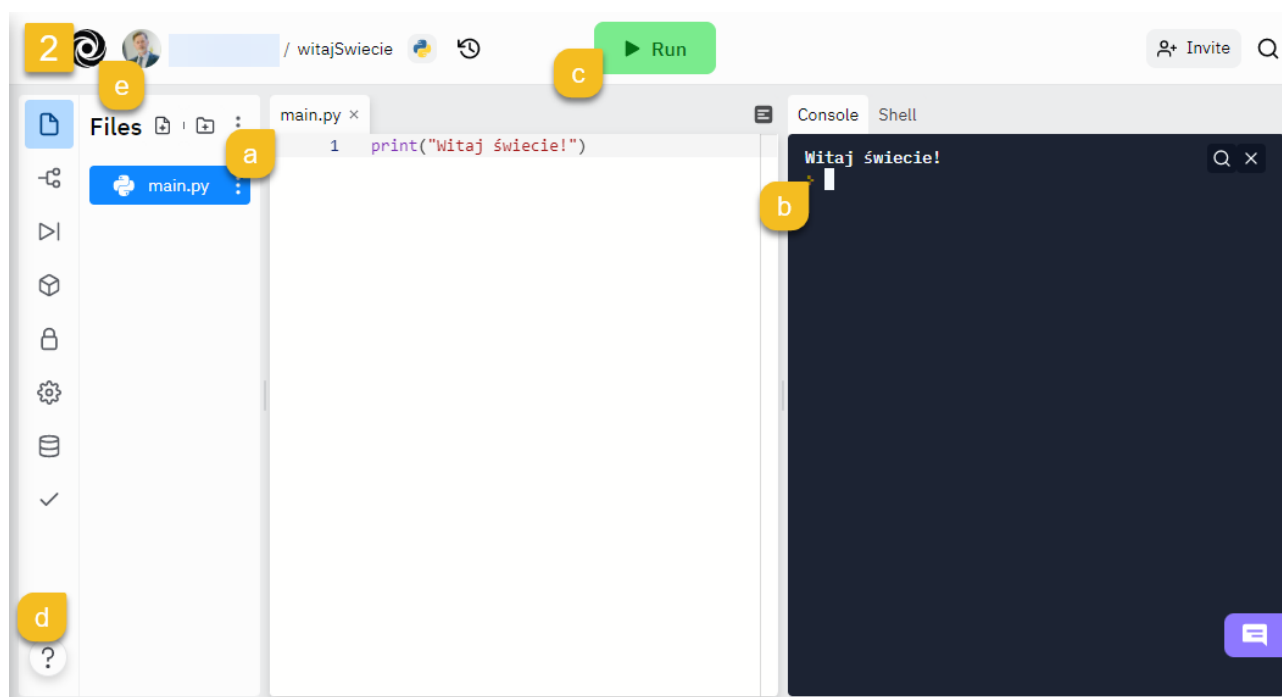


Rysunek 1. Okno tworzenia nowego projektu (autor Jerzy Dorożko, na podstawie zrzutu ekranu serwisu replit.com)

Po zalogowaniu otrzymują od nauczyciela wskazówki dotyczące podstaw korzystania z serwisu, po czym tworzą swój pierwszy projekt (Rysunek 1), wybierając polecenie **+ Create** [1], po czym z nowego menu wybierają szablon **Python** [a]. W przyszłości będą również korzystać z szablonu **Python (with Turtle)** [b]. Należy zwrócić uwagę na to, by projektowi nadali nazwę (np. **"Witaj świecie!"**) [c] oraz zatwierdzili zmiany kliknięciem **+ Create Repl** [d].

W kolejnym oknie (Rysunek 2) uczniowie otrzymują dostęp do **edytora skryptów** [a], których efekt działania dostępny będzie w oknie **konsoli** [b], po uruchomieniu przyciskiem **Run** [c]. Warto zwrócić uwagę uczniów na dostępne na pasku po lewej stronie **ikony poleceń**, w tym **Pomoc** dostępną pod znakiem zapytania [d].

Interfejs programu wymaga podstaw znajomości języka angielskiego. W razie problemów warto wspomóc się np. bezpłatną usługą Tłumacza Google. Należy jednak mieć na uwadze, że tłumaczeń należy dokonywać w osobnej karcie, metodą kopiuj/wklej. Korzystanie z usługi Tłumacza Google poprzez automatyczne tłumaczenie całej strony może prowadzić do nieoczekiwanych zachowań (błędów w działaniu).

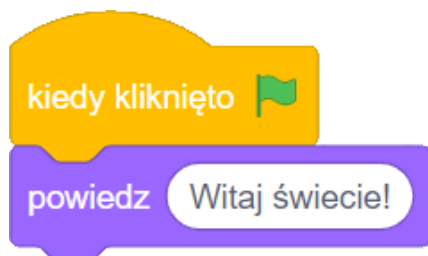


Rysunek 2. Interfejs edytora skryptu Replit (autor Jerzy Dorożko, na podstawie zrzutu ekranu serwisu replit.com)

Ćwiczenie 1:

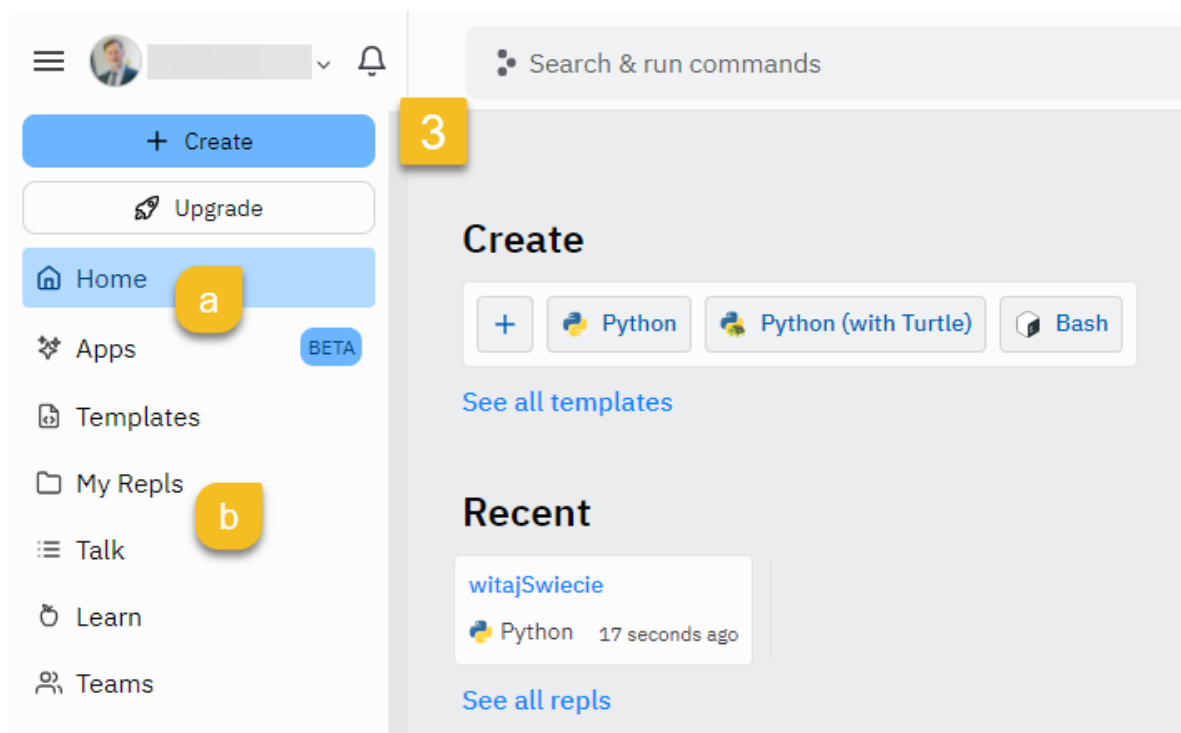
Przepisz poniższy kod programu w oknie edytora, uruchom go przyciskiem **Run** i zaobserwuj wynik działania w konsoli. Porównaj treść skryptu z programem o podobnym działaniu, napisanym w środowisku Scratch (Rysunek 3).

```
print("Witaj świecie!")
```



Rysunek 3. "Witaj świecie!" w środowisku Scratch (autor Jerzy Dorożko, na podstawie zrzutu ekranu serwisu scratch.mit.edu)

Po przeprowadzeniu i omówieniu powyższego ćwiczenia, uczniowie powinni przejść do **katalogu domowego** użytkownika, klikając jego charakterystyczne logo [e] (Rysunek 2). Utworzone projekty mogą tu organizować w **foldery** [b] (Rysunek 4). Z tego miejsca, w sekcji **Create**, będą zaczynać nowe projekty.



Rysunek 4. Katalog domowy użytkownika Replit (autor Jerzy Dorożko, na podstawie rzutu ekranu serwisu replit.com)

Kolejne ćwiczenia nauczyciel przeprowadza, zwracając uwagę uczniów na podobieństwa i różnice podczas programowania na przemian w obu środowiskach.

Ćwiczenie 2:

Zmienne w Scratchu i Pythonie. Nauczyciel prezentuje tworzenie zmiennej w Scratchu, zwracając uwagę uczniów na to, iż zmienna musi tu być utworzona wcześniej, jeszcze zanim zostanie do niej przypisana jakakolwiek wartość.



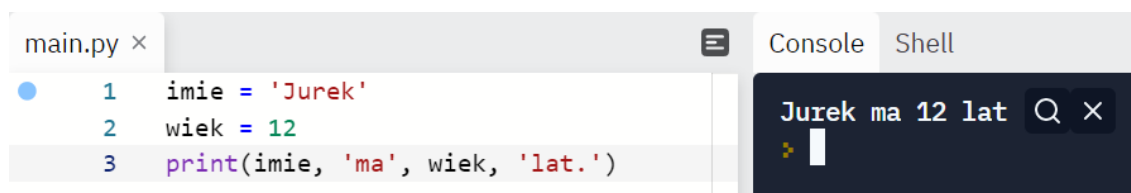
Rysunek 5. Zmienne w Scratchu (autor Jerzy Dorożko, na podstawie zrzutu ekranu serwisu scratch.mit.edu)

Uczniowie w oknie nowoutworzonego skryptu Pythona przepisują poniższy kod:

```
imie = 'Jurek'
```

```
wiek = 12
```

Od nauczyciela dowiadują się, że w Pythonie zmienna tworzona jest po przypisaniu wartości. Następnie uzupełniają kod skryptu i uruchamiają go.



Rysunek 6. Zmienne w Pythonie (autor Jerzy Dorożko, na podstawie zrzutu ekranu serwisu replit.com)

Dodatkowego komentarza ze strony nauczyciela będzie wymagała funkcja **print**, która zastąpiła w skrypcie bloczek **powiedz** ze Scratcha. Nauczyciel przedstawia obydwie jako instrukcje wyjścia.

EWALUACJA ZAJĘĆ:

Zadanie 1:

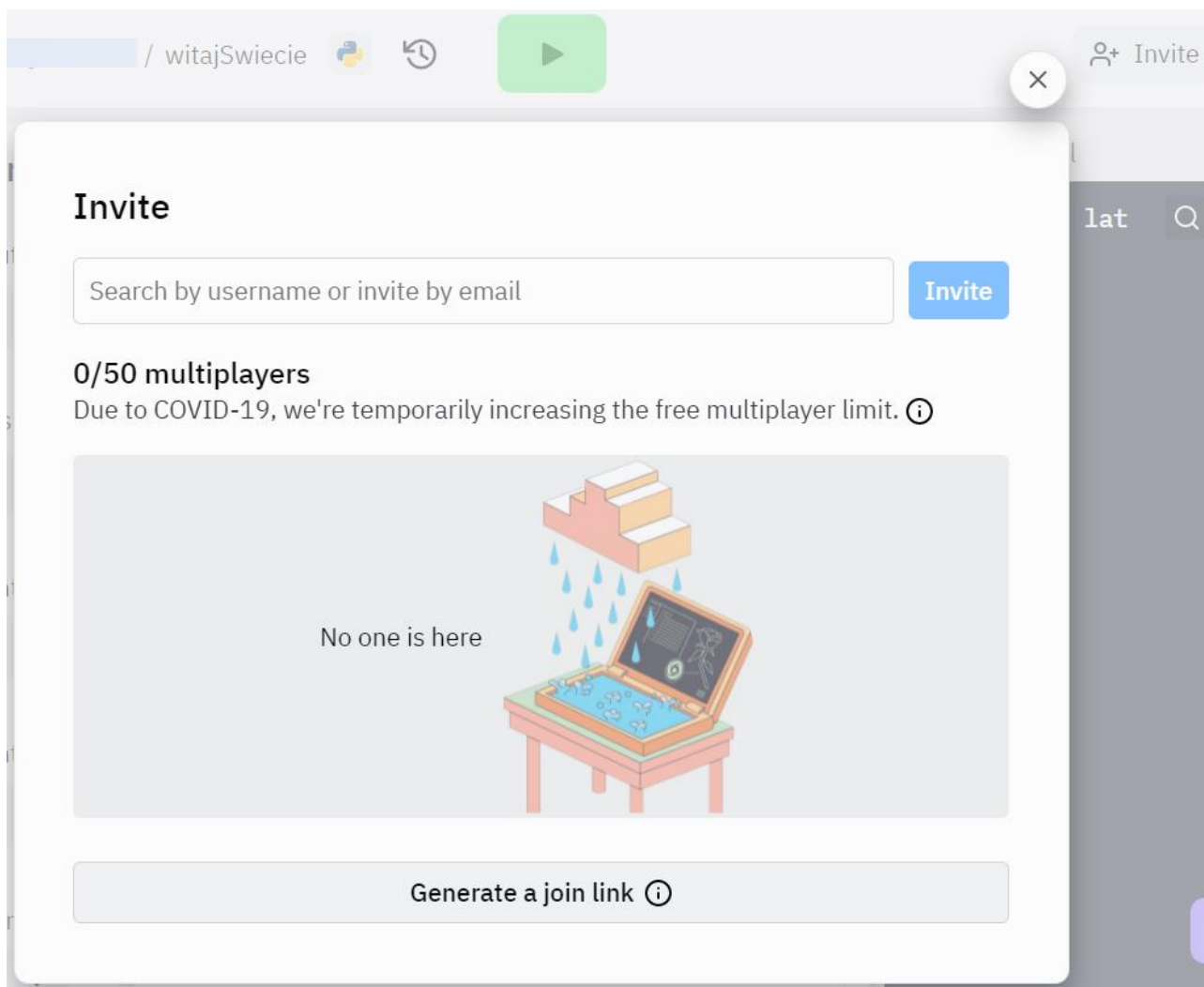
Nauczyciel prezentuje program utworzony w Scratchu i prosi uczniów, aby utworzyli jego odpowiednik w Pythonie.



Rysunek 7. Kod programu do zadania 1 (autor Jerzy Dorożko, na podstawie zrzutu ekranu serwisu scratch.mit.edu)

Zadanie 2:

Nauczyciel przedstawia dostępną w środowisku Replit możliwość zaproszenia innego użytkownika do współpracy nad projektem (Rysunek 8). Dzieli uczniów na dwuosobowe zespoły i prosi, aby zaprosili się nawzajem do realizowanego projektu. W kodzie projektu każdy z uczniów umieszcza błąd, który powinien zostać znaleziony i naprawiony przez drugiego członka zespołu.



Rysunek 8. Okno zaproszenia do projektu (autor Jerzy Dorożko, na podstawie zrzutu ekranu serwisu replit.com)

Wskazówki dla nauczyciela:

Opisane zajęcia można przeprowadzić zarówno w trybie stacjonarnym, jak i zdalnym. Uczniowie, którzy w klasach wcześniejszych zdobyli już doświadczenie w programowaniu z wykorzystaniem środowiska online (Scratch), nie powinni mieć problemów z możliwościami, jakie daje serwis replit.com.

Zajęcia zostały zaplanowane na 45 minut, jednak może okazać się, że w trybie online uczniowie będą potrzebować więcej czasu na zrozumienie wskazówek nauczyciela. W takiej sytuacji warto rozważyć rozłożenie przedstawionych powyżej treści na dwie jednostki metodyczne.

Ważnym elementem, który występuje dopiero w części końcowej lekcji, jest możliwość współpracy online w zespołach dwuosobowych. Należy poświęcić tej formie pracy należną uwagę i wykorzystywać ją w przyszłości.

Pozostałe podstawowe struktury programistyczne, które uczniowie poznali podczas nauki programowania w Scratchu, warto omawiać w przypadku Pythona w formie podobnej do wyżej przedstawionej. Przywołujemy przykłady ze środowiska wizualnego, po czym prezentujemy ich formę tekstową w Pythonie.

BIBLIOGRAFIA:

1. [Dokumentacja biblioteki Turtle w Pythonie](#) (serwis docs.python.org)
[dostęp: 01.02.22]
2. [Pomysły na projekty w Scratchu](#) (serwis scratch.mit.edu) [dostęp: 01.02.22]