

## Informatyka klasa 5

### Wymagania edukacyjne

#### DZIAŁ 1. Klawiatura zamiast pióra

##### 1.1 Dokumenty bez tajemnic (PP: II.3b)

- 2 — tworzy dokument z tytułem +  $\geq 2$  akapitami; używa pogrubienia i wyrównania; zapisuje plik.  
Dowody: plik .doc(x). Typowe błędy dozwolone: podwójne spacje, nadmiar „Enterów”.
- 3 — stosuje style (Nagłówek/Normalny), interlinię 1,15–1,5, wcięcie pierwszego wiersza; nadaje sensowną nazwę i miejsce zapisu.
- 4 — spójny układ (nagłówek/stopka, numeracja stron), korekta składu przez „Pokaż wszystko”; brak „sierot” w tytułach.
- 5 — hierarchia:  $\geq 2$  poziomy nagłówków, jednolita typografia (rozmiary, odstępy); świadome użycie skrótów klawiaturowych.
- 6 — tworzy mini-szablon (style, numeracja, marginesy) + 1-akapitową instrukcję korzystania (kiedy stosować które style).

##### 1.2 Kolejno odlicz! Style i numerowanie (PP: II.3b)

- 2 — wstawia listę punktowaną; nie miesza punktów z numeracją.
- 3 — stosuje listę numerowaną + Nagłówek 1/2; numeracja nie „łamie się” przy edycji.
- 4 — generuje spis treści ze stylów, aktualizuje go po zmianach; poprawne odstępy przed/po nagłówkach.
- 5 — konsekwentna hierarchia w całym pliku; unika ręcznego wstawiania numerów w tytułach.
- 6 — projekt szkic pracy dłuższej (sekcje, podsekcje + spis), z krótkim uzasadnieniem struktury.

##### 1.3 Komórki, do szeregu! Świat tabel (PP: II.3b, wsparcie planowania I.1a)

- 2 — wstawia tabelę min.  $3 \times 3$ , wpisuje dane; nie łamie nagłówków w dwóch wierszach bez potrzeby.
- 3 — formatuje nagłówek (pogrubienie), ustala wyrównanie i szerokość kolumn; nie rozciąga wierszy „na siłę”.
- 4 — dodaje tytuł i podpis tabeli; zachowuje czytelność (kontrast, odstępy).
- 5 — projektuje tabelę porównawczą (kryteria w kol.1), uzasadnia dobór kolumn; dba o jednostki i etykiety.
- 6 — tworzy mini-raport (tabela + 2–3 wnioski w tekście), wersja do druku.

##### 1.4 Nie tylko tekst. Wstawianie ilustracji (PP: II.3b, prawo V.2)

- 2 — wstawia 1 obraz, skaluje proporcjonalnie.

- 3 — ustawia zawijanie, pozycjonuje względem tekstu; zachowuje marginesy.
- 4 — dodaje podpis źródła (autor/licencja/link); zapewnia kontrast napisu względem tła.
- 5 — dobiera format (druk/ekran), nie pogarsza jakości (bez rozciągania bitmap).
- 6 — opracowuje checklistę podpisów (co najmniej: autor, tytuł/źródło, licencja, data dostępu).

#### 1.5 Przyrodnicze wędrówki. Atlas – projekt (PP: II.3b; IV.1–IV.2; chmura III.2a)

- 2 — oddaje swoją kartę (tytuł, tekst 5–8 zdań, 1 obraz).
- 3 — stosuje ustalone style zespołu; umieszcza plik w folderze projektu (współdzielonym).
- 4 — scala karty (spójne style, numeracja stron); dodaje spis kart.
- 5 — prezentuje wspólny dokument; wprowadza poprawki po feedbacku.
- 6 — rola redaktora: standard podpisów źródeł, porządek w folderach, publikacja „tylko do odczytu” + krótka notatka zmian.

### **DZIAŁ 2. Prawie jak w kinie. Prezentacje**

#### 2.1 Tekst i obraz. Najprostsza prezentacja (PP: II.3d, prawo V.2)

- 2 — 3 slajdy: tytuł/treść/podsumowanie; zapis pliku.
- 3 — dodaje obraz i listę punktowaną; zachowuje marginesy i czytelność.
- 4 — spójny szablon (czcionki, rozmiary, kolory), podpis źródła pod obrazem.
- 5 — notatki prelegenta do slajdów; ogranicza efekty (czytelność > „fajerwerki”).
- 6 — scenariusz wystąpienia (tezy + czas trwania), umie zsynchronizować mówienie z treścią slajdów.

#### 2.2 Wspomnienia z... Album fotograficzny (PP: II.3d; I.1a; III.1a–b)

- 2 — układa ciąg slajdów ze zdjęciami.
- 3 — dopasowuje układ/rozmiary zdjęć; krótkie opisy.
- 4 — używa siatki/wytocznych do równego rozmieszczenia; zachowuje proporcje.
- 5 — eksport pokazu (format, rozmiar) i test odtwarzania.
- 6 — przygotowuje intro/outro (tytuł, autorzy, źródła), uzasadnia wybór układu.

#### 2.3 Wprawić świat w ruch. Przejścia i animacje (PP: II.3d)

- 2 — proste przejścia między slajdami.
- 3 — animacja wejścia elementu.
- 4 — kolejność i czasy animacji zgodne z logiką przekazu.
- 5 — zachowana czytelność (bez przeładowania efektami), powtarzalny rytm.

- 6 — porównuje dwa warianty, argumentuje wybór lepszego (cel, grupa odbiorców).

#### 2.4 Dźwięk i wideo w prezentacji (PP: II.3d; III.1a–b, prawo V.2)

- 2 — wstawia dźwięk/wideo.
- 3 — ustawia autoodtwarzanie/po kliknięciu, głośność.
- 4 — przycina klip (początek/koniec), dodaje opis źródła/licencji.
- 5 — dobiera format/rozmiar plików; sprawdza zgodność na innym komputerze.
- 6 — tworzy zwiastun, zachowuje prawa do materiałów.

#### 2.5 Krótka historia. Sterowanie animacją (PP: II.3d)

- 2 — opowieść  $\geq 4$  slajdy (początek–rozwinięcie–koniec).
- 3 — proste wyzwalacze/czasy.
- 4 — synchronizuje tekst/obrazy/przejścia z narracją.
- 5 — hiperłącza/menu do nawigacji; struktura przejrzysta.
- 6 — prezentacja z komentarzem ustnym + wersja dla osób nieobecnych.

### **DZIAŁ 3. Kocie sztuczki. Scratch**

#### 3.1 Plan to podstawa. Rozwiązywanie problemów (PP: I.3)

- 2 — zapisuje kroki zadania w kolejności (lista).
- 3 — dopisuje kryterium sukcesu („działa, gdy...”).
- 4 — dzieli problem na podzadania; wskazuje miejsce testu.
- 5 — porównuje 2 sposoby; wybiera i uzasadnia wybór.
- 6 — tworzy pseudokod/schemat blokowy + krótko prezentuje zespołowi.

#### 3.2 W poszukiwaniu skarbu. Labirynt (PP: II.1a; II.1b; II.2; I.2b)

- 2 — steruje duszkiem klawiszami; uruchamia/zatrzymuje skrypt.
- 3 — buduje sekwencję + pętlę ruchu; stosuje sensowny krok.
- 4 — dodaje warunek (ściana→zawrót), testuje i poprawia błąd.
- 5 — projektuje poziom (cel/zasady), dostraja prędkość/rotację do grywalności.
- 6 — przygotowuje instrukcję użytkownika (sterowanie, cel, restart), nagrywa krótkie demo.

#### 3.3 Scena niczym kartka. Rysowanie (PP: II.1a; II.2)

- 2 — rysuje kształt sekwencją kroków.
- 3 — stosuje pętlę do powtarzania rysunku.
- 4 — uruchamia rysowanie zdarzeniem.

- 5 — opisuje algorytm własnymi słowami; testuje różne parametry.
- 6 — porównuje dwa skrypty osiągające ten sam efekt i wybiera bardziej zwięzły.

### 3.4 Od wielokąta do rozety (PP: II.1a; II.2)

- 2 — tworzy wielokąt pętłą.
- 3 — reguluje liczbę boków/kąt; nie „rozjeżdża” się rysunek.
- 4 — składa rozetę (powtarzanie + obrót).
- 5 — komentuje bloki, zapisuje parametry (np. boki=6, kąt=60).
- 6 (rozszerzenie) — dodaje zmienną sterującą parametrem (np. boki).

## **DZIAŁ 4. Bieganie po ekranie. Pivot Animator**

### 4.1 Patyczaki w ruchu. Proste animacje (PP: II.3a)

- 2 — tworzy ciąg klatek prostej animacji.
- 3 — reguluje kl./s dla płynności; unika „teleportów” między klatkami.
- 4 — utrzymuje spójny ruch (start/stop, kierunek), dobiera tło.
- 5 — eksport GIF/MP4 w sensownym rozmiarze; sprawdza odtwarzanie.
- 6 — przygotowuje storyboard (3–4 ujęcia) i realizuje zgodnie z planem.

### 4.2 Animacje od kuchni. Własne postaci (PP: II.3a)

- 2 — modyfikuje gotową postać (pozycja/rozmiar).
- 3 — projektuje szkielet postaci (segmenty), zapisuje warianty.
- 4 — nadaje charakter ruchu (zgięcia, rotacje); zachowuje proporcje.
- 5 — składa postać + tło + podpisy.
- 6 — tworzy biblioteczkę  $\geq 5$  pozycji dla zespołu (stand, bieg, skok, upadek, pauza).

### 4.3 Podróż z przeszkodami. Film przygodowy – projekt (PP: II.3a; II.3d; IV.1–IV.2)

- 2 — oddaje własną scenę (krótka sekwencja, zgodna z konwencją projektu).
- 3 — dotrzymuje harmonogramu; nazwy/formaty plików zgodne ze standardem zespołu.
- 4 — montuje przejścia/napisy; zapewnia rytm i czytelność.
- 5 — film spójny narracyjnie (początek–rozwińcie–zakończenie), podpisane źródła.
- 6 — rola koordynatora: plan, podział ról, publikacja „wersji do pokazu”, krótka retrospekcja (co działa/co poprawić).