

Log Book  
Game Tebak Nama Hewan

Tanggal & Waktu	AI yang digunakan	Prompt / Pertanyaan	Respon AI (Ringkasan Kode)	Implementasi dan Refleksi tim
11/11/2025 11:40	Chatgpt	kamu adalah seorang ahli, buatlah saya game tebak kata menggunakan python tanpa menggunakan library lain seperti import math, import random dll, buatlah tampilan yang bagus, buatlah dengan tema	<pre># tebak_kata.py # Game Tebak Kata — tanpa import eksternal # Jalankan dengan Python 3 (terminal) RESET = "\033[0m" BOLD = "\033[1m" FG_CYAN = "\033[36m" FG_GREEN = "\033[32m" FG_RED = "\033[31m" FG_YELLOW = "\033[33m" FG_MAG = "\033[35m"  HANGMAN_STAGES = [ r''' +---+                               ===== r'''</pre>	<p>Implementasi: Tim mencoba menjalankan kode game tebak kata sesuai instruksi AI.</p> <p>Refleksi: Kode yang diberikan AI sangat membantu mempercepat pengembangan, terutama karena tidak perlu library tambahan.</p>

			<pre> +---+     O         ===== r''' </pre>	
12/11/2025 11:25	Chatgpt	<p>saya mau diatas soal ada tebak hewan seperti misal tikus menjadi "T_U_" dan jika ada yang mengetik bantuan nambah 1 huruf menjadi "TI_U_" bantuan terdapat 3, jika bantuan sudah memenuhi jawaban akan dianggap</p> <p>tidak valid</p>	<pre> ''' 🐾 Game Tebak Kata: HEWAN (Versi Bantuan &amp; Huruf Terbuka) ===== =====  Fitur: - Tidak pakai import apapun (pure Python). - Soal muncul seperti "T_U_" (huruf pertama terlihat). - Bisa minta 'bantuan' maksimal 3x → buka huruf tambahan. - Kalau bantuan sudah membuka semua huruf, dianggap gagal otomatis. - Ada hint lucu &amp; feedback gokil. '''  # ----- DATA HEWAN ----- HEWAN_LIST = [     {"nama": "harimau", "habitat": "hutan", "ukuran": "besar", "makanan": "daging", "suara": "auman"},     {"nama": "hiu", "habitat": "laut", "ukuran": "besar", "makanan": "ikan", "suara": "diam tapi seram"}, </pre>	<p>Implementasi: Struktur permainan berhasil dibuat: tampilan awal, proses input huruf, tampilan hangman, dan kondisi menang/kalah berjalan sebagaimana mestinya.</p> <p>Refleksi: Beberapa bagian kode perlu sedikit penyesuaian manual agar gaya tampilan (warna/hiasan ASCII) cocok dengan</p>

			<pre> {"nama": "singa", "habitat": "sabana", "ukuran": "besar", "makanan": "daging", "suara": "raaawwrr"}, {"nama": "kucing", "habitat": "rumah", "ukuran": "kecil", "makanan": "ikan", "suara": "meong"}, {"nama": "tikus", "habitat": "ladang", "ukuran": "kecil", "makanan": "biji", "suara": "ciit"}, {"nama": "cicak", "habitat": "dinding", "ukuran": "kecil", "makanan": "serangga", "suara": "klik klik"}, {"nama": "gajah", "habitat": "hutan", "ukuran": "raksasa", "makanan": "rumput", "suara": "truum"}, {"nama": "kelinci", "habitat": "padang", "ukuran": "kecil", "makanan": "wortel", "suara": "diam manis"}, {"nama": "burung", "habitat": "udara", "ukuran": "kecil", "makanan": "biji", "suara": "kicau"}, {"nama": "kura-kura", "habitat": "air/tanah", "ukuran": "sedang", "makanan": "daun", "suara": "diam tapi sabar"}, ]  # ----- PEMILIH HEWAN TANPA IMPORT ----- def pseudo_random_index(nama_pemain, panjang):     """Hasil pseudo-random dari nama pemain (tanpa library)."""     total = 0     for ch in nama_pemain:         total += ord(ch)     return (total * len(nama_pemain)) % panjang  # ----- HINT ----- </pre>	perangkat yang digunakan tim.
--	--	--	--	-------------------------------

			<pre>def hint_bertingkat(hewan, tahap):     if tahap == 1:         return f"Hewan ini punya {len(hewan['nama'])} huruf, dan suka di {hewan['habitat']}."     elif tahap == 2:         return f"Makanannya biasanya {hewan['makanan']}."     elif tahap == 3:         return f"Ukuran tubuhnya {hewan['ukuran']}."     elif tahap &gt;= 4:         return f"Suaranya: '{hewan['suara']}'."     else:         return "Hint habis! Coba pakai logika hewanmu sendiri."</pre>	
16/11/2025 10.44	Chatgpt	coba tanpa feedback yang lucu. Misalnya, kalau tebakan terlalu rendah di mode Hewan Tebakanmu sekecil tikus di bawah meja hilangkan biar orang yang memainkannya tidak pusing karena banyak teks	<pre>""" 🐾 Game Tebak Kata: HEWAN (Versi Simpel, Tanpa Feedback Lucu) ===== ===== - Tidak pakai import apapun (pure Python). - Soal muncul seperti "T__U_". - Bisa minta 'bantuan' maksimal 3x (buka 1 huruf tiap kali). - Kalau semua huruf terbuka, dianggap tidak valid. - Ada hint sederhana.</pre>	<p>Implementasi: Fitur bantuan (membuka huruf satu per satu) juga diuji dan bekerja sesuai permintaan, dengan batas 3 kali bantuan.</p> <p>Refleksi:</p>

			<pre> """ # ----- DATA HEWAN ----- HEWAN_LIST = [     {"nama": "harimau", "habitat": "hutan", "ukuran": "besar", "makanan": "daging", "suara": "auman"},     {"nama": "hiu", "habitat": "laut", "ukuran": "besar", "makanan": "ikan", "suara": "diam"},     {"nama": "singa", "habitat": "sabana", "ukuran": "besar", "makanan": "daging", "suara": "auman"},     {"nama": "kucing", "habitat": "rumah", "ukuran": "kecil", "makanan": "ikan", "suara": "meong"},     {"nama": "tikus", "habitat": "ladang", "ukuran": "kecil", "makanan": "biji", "suara": "ciit"},     {"nama": "cicak", "habitat": "dinding", "ukuran": "kecil", "makanan": "serangga", "suara": "klik"},     {"nama": "gajah", "habitat": "hutan", "ukuran": "besar", "makanan": "rumput", "suara": "truum"},     {"nama": "kelinci", "habitat": "padang", "ukuran": "kecil", "makanan": "sayur", "suara": "diam"},     {"nama": "burung", "habitat": "udara", "ukuran": "kecil", "makanan": "biji", "suara": "kicau"},     {"nama": "kura-kura", "habitat": "air/tanah", "ukuran": "sedang", "makanan": "daun", "suara": "diam"}, ]  # ----- PEMILIH HEWAN TANPA IMPORT ----- </pre>	<p>Tim menyadari pentingnya memahami alur logika permainan (loop tebakan, validasi input, dan pengecekan huruf).</p>
--	--	--	---	--

			<pre>def pseudo_random_index(nama_pemain, panjang):     """Menghasilkan indeks pseudo-random dari nama pemain."""     total = 0     for ch in nama_pemain:         total += ord(ch)     return (total * len(nama_pemain)) % panjang</pre>	
17/11/2025 11.49	Chatgpt	<p>okeh, semuanya benar, tapi aku mau ada 3 soal atau lebih, jika soal pertama benar maka mempunyai 1 point, dstnya, dan aku mau mempunyai 3 kesempatan untuk menjawab, jika ketiga soal tersebut ada yang salah maka point akhirnya dikurang yang salah</p>	<pre>""" 🐾 Game Tebak Kata: HEWAN (Versi Multi Soal + Skor) ===== ===== - Tanpa import apa pun (pure Python). - Ada 3 soal (bisa ditambah mudah). - Tiap jawaban benar = +1 poin. - Tiap jawaban salah = -1 poin (maksimal 3 kesalahan = game over). - Bantuan maksimal 3x per soal (buka huruf satu per satu). - Kalau semua huruf terbuka → dianggap tidak valid. """  # ----- DATA HEWAN ----- HEWAN_LIST = [     {"nama": "harimau", "habitat": "hutan", "ukuran": "besar",      "makanan": "daging", "suara": "auman"},     {"nama": "hiu", "habitat": "laut", "ukuran": "besar", "makanan":      "ikan", "suara": "diam"},     {"nama": "singa", "habitat": "sabana", "ukuran": "besar",      "makanan": "daging", "suara": "auman"},</pre>	<p>Implementasi: Tampilan antarmuka berbasis terminal sudah cukup jelas dan mudah dipahami pemain.</p> <p>Refleksi: Fitur bantuan terbukti meningkatkan pengalaman bermain, tetapi harus dibatasi agar permainan tetap menantang.</p>

			<pre> {"nama": "kucing", "habitat": "rumah", "ukuran": "kecil", "makanan": "ikan", "suara": "meong"}, {"nama": "tikus", "habitat": "ladang", "ukuran": "kecil", "makanan": "biji", "suara": "ciit"}, {"nama": "cicak", "habitat": "dinding", "ukuran": "kecil", "makanan": "serangga", "suara": "klik"}, {"nama": "gajah", "habitat": "hutan", "ukuran": "besar", "makanan": "rumpun", "suara": "truum"}, {"nama": "kelinci", "habitat": "padang", "ukuran": "kecil", "makanan": "sayur", "suara": "diam"}, {"nama": "burung", "habitat": "udara", "ukuran": "kecil", "makanan": "biji", "suara": "kicau"}, {"nama": "kura-kura", "habitat": "air/tanah", "ukuran": "sedang", "makanan": "daun", "suara": "diam"}, ]  # ----- PEMILIH HEWAN TANPA IMPORT ----- def pseudo_random_index(nama_pemain, panjang, offset=0):     """Menghasilkan indeks pseudo-random dari nama pemain."""     total = 0     for ch in nama_pemain:         total += ord(ch)     return (total * len(nama_pemain) + offset * 7) % panjang </pre>	
--	--	--	---	--

18/11/2025 11.10	Chatgpt	<p>kamu adalah ahli, Tolong buat aku code Python. game text-based yang harus lucu dan compatible Temanya: Game 'Tebak Kata' HEWAN . Detail Game: Di awal, game HEWAN (misalnya, hewan rahasia harimau, hiu, singa, ku cing, tikus, cicak dsb, Hints-nya terkait Hewan tersebut) Hints Lucu: Setelah pemain</p>	<pre># Tebak Kata - HEWAN (lucu) # Compatible Python 3, TANPA import eksternal sama sekali. # Penjelasan singkat: # - Game memilih satu hewan dari daftar berdasarkan "seed" (kata acak dari pemain). # - Tampilkan pola seperti "T____" (huruf pertama + underscore). # - Pemain menebak sampai benar atau mengetik "quit". # - Ketik "hint" untuk minta petunjuk (ada beberapa petunjuk, dibatasi). # - Setelah setiap tebakan, diberikan feedback lucu berdasarkan kemiripan tebakan.  # === DATA HEWAN: setiap hewan punya nama (kata kunci) dan beberapa hint lucu === animals = [     {         "name": "harimau",</pre>	<p>Implementasi: Kata-kata hewan diuji beberapa kali (contoh: "tikus", "ular", "gajah") dan semua berjalan tanpa error.</p> <p>Refleksi: Untuk pengembangan selanjutnya, tim berencana menambahkan pilihan kategori hewan, skor, dan riwayat permainan.</p>



		<p>menebak, kasih feedback yang lucu. Misalnya, kalau tebakan terlalu rendah di mode Hewan Tebakanmu sekecil tikus di bawah meja</p> <p>Gak Pakai Ribet: NO LIBRARY EKSTERNAL SAMA SEKALI. Cuma boleh pakai modul bawaan Python(jangan ada import random, import math dsb)</p> <p>Error: Kode WAJIB jalan mulus 100% dan compatible Python 3.</p> <p>Komentar: Kasih keterangan di kodenya biar aku gak bingung. masukan juga dibawah soal misal "T____"</p> <p>dibawahnya ada hint otomatisnya, dan jika ada yang meminta hint beri hint/ciri-ciri lain,</p>	<pre> "hints": [     "Saya bergaris — seperti pisang yang lari dari supermarket.",     "Keluarga besar kucing, tapi nggak mau dipeluk.",     "Biasanya raja hutan... kalau nggak sedang tidur siang." ] }, {     "name": "hiu",     "hints": [         "Hidup di laut, giginya rapih dan nggak pernah bilang 'maaf'.",         "Biasanya terlihat saat kau sengaja membuang roti ke laut.",         "Bukan ikan kucing, tapi suka menonton film thriller."     ] }, {     "name": "singa",     "hints": [         "Rambutnya bergaya, kadang disebut 'mewah'.",         "Raja padang rumput (jika padang rumputnya mau jadi kerajaan).",         "Suara kerasnya sering dipakai untuk alarm bangun tidur."     ] } </pre>	
--	--	---	---	--

		jadi hint ada banyak dan hint tetap dibatasi	<pre> "name": "kucing", "hints": [   "Suka tidur 20 jam sehari dan mengatur rumah (dari sofa).",   "Jika kamu salah menaruh makanan, dia akan mengoreksi dengan tatapan.",   "Suara: 'meong' (kadang matinya main petak umpet)."] }, {   "name": "tikus",   "hints": [     "Kecil, gesit, dan suka kejutan di gudang.",     "Suka keju (katanya), tapi kadang lebih suka roti.",     "Bisa menari di labirin tanpa GPS."   ] }, {   "name": "cicak",   "hints": [     "Menempel di dinding seperti poster yang hidup.",     "Kalau jatuh kadang pura-pura mau kabur.",     "Suara kecilnya sering muncul tengah malam: 'cek-cek-cek'."   ] } </pre>	
--	--	--	---	--

			<pre> "name": "gajah", "hints": [     "Besar, ingatannya kuat — jangan beri dia hutang.",     "Punya belalai yang multifungsi: mandi, minum, dan selfie.",     "Suara gemuruhnya sering membuat cangkir bergetar." ] }, {     "name": "kelinci",     "hints": [         "Berlompat-lompat seperti kue yang bahagia.",         "Suka wortel (menurut brosur nya).",         "Telinganya panjang; barang hilang biasanya ditemui di telinga dia."     ] } ] </pre>	
18/11/2025 11.20	Chatgpt	kamu adalah seorang ahli, buatlah saya game tebak nama hewan menggunakan python 3 fokus dengan logika program yang kuat, alur yang jelas, dan interaksi teks yang	<pre> # 🎮 Game Tebak Nama Hewan - Dibuat oleh ChatGPT (GPT-5) # Hanya menggunakan Python Standar Library # Tema: Hewan 🐯🐸🐊  import random import time import os </pre>	Implementasi: Tim melakukan uji coba permainan dengan beberapa pengguna lain untuk memastikan input huruf dan sistem

		<p>intuisif, hanya boleh memakai bawaan python atau standart library dan jangan instal library dari luar(pip install). contoh numpy pandas pygame dan sebagainya, buat tampilan yang bagus dan menarik, dengan tema tebak nama hewan, mekanisme nya ada 5 soal, diatas soal terdapat peraturan dan cara main game tebak nama hewanya, dan saya mau soal nya ada beberapa yang ada dan tidak ada misal "T_K__" dan contoh lain kuncinya jangan rumit, yang penting solufit dan orisinal</p>	<pre># ===== # Tampilan Awal &amp; Aturan # ===== def tampil_judul():     os.system('cls' if os.name == 'nt' else 'clear')     print("=" * 50)     print("🐾 SELAMAT DATANG DI GAME TEBAK NAMA HEWAN 🐾".center(50))     print("=" * 50)     print("\n🎯 Peraturan &amp; Cara Main:")     print("- Akan ada 5 hewan yang harus kamu tebak.")     print("- Kamu akan diberi petunjuk berupa nama sebagian (misal: T_K__).")     print("- Jawab dengan nama hewan yang kamu pikir benar.")     print("- Setiap jawaban benar = +20 poin. Jawaban salah = 0 poin.")     print("- Semangat! Dan jangan ngasal kayak tebak mantan 😊\n")     input("Tekan ENTER untuk mulai... 🐾 ")  # ===== # Daftar Soal # ===== soal_hewan = [     {"soal": "T_K__", "jawaban": "TIKUS", "hint": "Suka mencuri makanan di dapur 🍳"},</pre>	<p>validasi bekerja konsisten.</p> <p>Kesalahan pengetikan (misal input angka atau huruf lebih dari satu) sudah ditangani dengan baik melalui pesan error yang ramah pengguna.</p> <p>Refleksi: Tim menyadari bahwa pemilihan kata hewan sangat berpengaruh pada tingkat kesulitan, sehingga dibutuhkan daftar kata yang seimbang antara hewan mudah dan hewan yang jarang dikenal.</p> <p>Terdapat wacana untuk menambahkan</p>
--	--	--	--	--

			<pre> {"soal": "K_C_NG", "jawaban": "KUCING", "hint": "Lucu, suka mengeong 🐱"}, {"soal": "_ARIM_U", "jawaban": "HARIMAU", "hint": "Raja hutan versi Asia 🐅"}, {"soal": "B_RUNG", "jawaban": "BURUNG", "hint": "Suka berkicau dan bisa terbang 🐦"}, {"soal": "_GA_", "jawaban": "SINGA", "hint": "Raja hutan versi Afrika 🦁"}, {"soal": "K_D_", "jawaban": "KODOK", "hint": "Melompat di rawa dan suka bilang 'kruk-kruk' 🐸"}, {"soal": "I_K_N", "jawaban": "IKAN", "hint": "Tinggal di air, bukan di hati kamu 🐟"}, {"soal": "_ELALANG", "jawaban": "BELALANG", "hint": "Jago lompat, suka di sawah 🦨"}, {"soal": "G_RAF_", "jawaban": "GIRAFa", "hint": "Lehernya panjang banget 🦒"}, {"soal": "A_YM", "jawaban": "AYAM", "hint": "Bangun pagi dan berkokok 'kukuruyuk!' 🐔"}, ] </pre>	level permainan (mudah–sedang–sulit) agar game bisa digunakan untuk anak-anak maupun pengguna umum.
18/11/2025 11.50	Chatgpt	Kamu adalah seorang ahli, buatlah saya game tebak nama hewan menggunakan python 3 fokus dengan logika program yang kuat, alur yang jelas,	<pre> import random import time import sys  def clear_screen():     """Membersihkan terminal (untuk tampilan yang lebih     baik)."""     # Untuk Windows     if sys.platform.startswith('win'):         import os         os.system('cls')     # Untuk Linux/macOS     else:         import os         os.system('clear')  def generate_soal(nama_hewan):     """Membuat format soal (tebakan) dari nama hewan. Beberapa     huruf akan diganti dengan underscore (_)."""     panjang = </pre>	Implementasi: Tim melakukan beberapa kali pengujian kata hewan acak (contoh: tikus, gajah, ular, ayam) untuk

		<p>dan interaksi teks yang intuitif, hanya boleh memakai bawaan python atau standart library dan jangan menginstal library dari luar(pip install). contoh numpy pandas pygame dan sebagainya, buatkan tampilan yang bagus dan menarik, dengan tema tebak nama hewan, mekanisme nya ada 5 soal, diatas soal terdapat peraturan dan cara main game tebak nama hewanya, dan saya mau soal nya ada beberapa yang ada dan tidak ada misal "T_K__" dan contoh lain kuncinya jangan rumit, yang penting solufit dan orisinal</p>	<pre>len(nama_hewan) # Jumlah huruf yang akan disembunyikan (minimal 1/3 dari panjang) jumlah_sembunyi = max(1, panjang // 3) # Pilih indeks secara acak untuk disembunyikan indeks_sembunyi = random.sample(range(panjang), jumlah_sembunyi) soal_list = list(nama_hewan.upper()) for i in indeks_sembunyi: soal_list[i] = '_' return "".join(soal_list)</pre>	<p>memastikan mekanisme tebakan dan validasi berjalan konsisten.</p> <p>Refleksi: Proses debugging berjalan lebih cepat karena alur permainan dibuat sederhana dan mudah dipahami.</p>
--	--	---	---	--

18/11/2025 11.55	Gemini AI	nah udah bagus, sekarang aku mau ada levelnya misal mudah sedang sulit, jika mudah huruf yang ditutup sedikit, jika sedang huruf yang ditutup menjadi lebih banyak dan jika sulit sedikit huruf yang di buka, dan aku mau juga ada hint bantuan dan quit untuk keluar dari permainan	<pre>def clear_screen(): """Membersihkan terminal (untuk tampilan yang lebih baik).""" # Untuk Windows if sys.platform.startswith('win'): import os os.system('cls') # Untuk Linux/macOS else: import os os.system('clear') def tampilkan_header(): """Menampilkan judul dan header game.""" print("=" * 60) print(f" {COLOR_HEADER} 🌟 GAME TEBAK NAMA HEWAN 🌟 {COLOR_RESET}") print("=" * 60) def tampilkan_aturan(): """Menampilkan peraturan dan cara bermain.""" tampilkan_header() print(f"\n{COLOR_HEADER} 📖 PERATURAN &amp; CARA BERMAIN{COLOR_RESET}") print("-" * 30) print(f"1. Anda akan memilih level dan memainkan \033[93m5 SOAL\033[0m.") print(f"2. Jawab dengan mengetikkan NAMA HEWAN LENGKAP.") print(f"3. Ketik '{COLOR_HIGHLIGHT}HINT{COLOR_RESET}' untuk meminta petunjuk (hanya 1x per soal).") print(f"4. Ketik '{COLOR_HIGHLIGHT}QUIT{COLOR_RESET}' untuk keluar dari</pre>	<p>Implementasi: Dari evaluasi sistem permainan, Gemini menilai bahwa alur interaksi sudah cukup intuitif. Namun, integrasi pesan kesalahan (error handling) bisa lebih diperkaya dengan memberikan rekomendasi langsung kepada pemain tentang langkah berikutnya.</p> <p>Refleksi:</p>

			<pre> permainan kapan saja.") print("5. Jawaban tidak bersifat *case-sensitive*.\n") print("-" * 60) input(f"Tekan {COLOR_SUCCESS}ENTER{COLOR_RESET} untuk melanjutkan...") clear_screen() </pre>	<p>Menurut analisis Gemini, game memiliki potensi edukasi yang baik, terutama dalam memperkenalkan kosakata nama hewan.</p> <p>Pengembangan ke depannya dapat mempertimbangkan penambahan ilustrasi atau mode pembelajaran untuk meningkatkan engagement pengguna.</p>
19/11/2025 11.00	Gemini AI	<p>buat skor point yangimbang, jika 5 soal maka point yang didapat per jawaban benar adalah 20, jika 10 soal maka per jawaban mendapatkan 10 point, level mudah 5 soal, level sedang 10</p>	<pre> # --- FUNGSI PENGANTI BUILT-IN --- def simple_rand(seed): """ Fungsi pengacakan sederhana (Pseudorandom Number Generator) menggantikan import random. """ # Linear Congruential Generator (LCG) a = 1664525 c = 1013904223 m = 2**32 seed = (seed % m + m) % m seed = (a * seed + c) % m return seed, seed / m def simple_shuffle_and_choice(data, count, seed): """ Mengacak list 'data' dan mengambil 'count' elemen, menggantikan random.shuffle dan random.sample. """ n = len(data) if count &gt; n: count = n temp_list = list(data) current_seed = seed # Algoritma Fisher-Yates shuffle </pre>	<p>Implementasi: Analisis Gemini menunjukkan bahwa struktur logika game sudah cukup baik, terutama dalam bagian validasi input dan kondisi menang-kalah.</p>



		beserta level sulit 10 soal	yang dimodifikasi for i in range(n - 1, 0, -1): current_seed, rand_val = simple_rand(current_seed) j = int(rand_val * (i + 1)) temp_list[i], temp_list[j] = temp_list[j], temp_list[i] return current_seed, temp_list[:count] def clear_screen_simple(): """Menggantikan import os.system('clear') dengan mencetak banyak baris kosong.""" print('\n' * 50)	Refleksi: <b>Menurut Gemini,</b> game memiliki potensi edukatif karena dapat mengkenalkan kosakata hewan sambil melatih logika pemain.



