|  |  |
| --- | --- |
| **Nama** | AGUNG IBRATAMA |
| **Tanggal Test** | 20 JUNI 2025 |

# **Psikotest DISC**

|  |  |  |  |
| --- | --- | --- | --- |
| **CONTOH** | **S** | **TS** |  |
|  | X |  | Ramah. Mudah bergaul |
|  |  |  | Penuh kepercayaan.Percaya pada orang lain |
|  |  | X | Petualang. Pengambil resiko |
|  |  |  | Toleran. Penuh hormat |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **S** | **TS** |  | **S** | **TS** |  | **S** | **TS** |
| **1.** |  | x | **2.** |  | x | **3.** | x |  |
|  |  | x |  |  |  |
| x |  |  |  |  | x |
|  |  |  |  |  |  |
| **4.** | x |  | **5.** |  | x | **6.** |  |  |
|  |  | x |  |  | x |
|  | x |  |  |  |  |
|  |  |  |  | x |  |
| **7.** |  | x | **8.** |  |  | **9.** | x |  |
|  |  |  | x |  |  |
|  |  |  |  |  | x |
| x |  | x |  |  |  |
| **10.** |  | x | **11.** |  |  | **12.** |  |  |
|  |  |  | x |  |  |
|  |  | x |  |  | x |
| x |  |  |  | x |  |
| **13.** |  |  | **14.** | x |  | **15.** |  | x |
|  | x |  | x |  |  |
|  |  |  |  |  |  |
| x |  |  |  | x |  |
| **16.** | x |  | **17.** |  |  | **18.** |  | x |
|  | x | x |  |  |  |
|  |  |  |  |  |  |
|  |  |  | x | x |  |
| **19.** | x |  | **20.** |  |  | **21.** |  |  |
|  | x | x |  | x |  |
|  |  |  | x |  |  |
|  |  |  |  |  | x |
| **22.** |  |  | **23.** |  |  | **24.** | x |  |
|  |  |  |  |  |  |
| x |  |  | x |  |  |
|  | x | x |  |  | x |

# SOAL LOGIKA

|  |  |
| --- | --- |
| Soal | Jawaban ( A, B, C, D ) |
| Soal 1 | E |
| Soal 2 | C |
| Soal 3 | D |
| Soal 4 | C |
| Soal 5 | E |

# Soal Programming

|  |  |
| --- | --- |
| **Soal Nomor** | 1 |
| **Jawaban Soal** | function hitungTinggi(tinggiAwal, lamaHari, pertumbuhan) {    let tinggi = tinggiAwal;    for (let i = 0; i < lamaHari; i++) {      tinggi = tinggi \* (1 + pertumbuhan);    }    return 'Tinggi akhir:' + tinggi.toFixed(2);  }  console.log(hitungTinggi(200, 5, 0.05));  console.log(hitungTinggi(250, 2, 0.02)); |
| ScreenShot |  |

|  |  |
| --- | --- |
| **Soal Nomor** | 2 |
| **Jawaban Soal** | function tangga(tinggi) {    for (let i = 1; i <= tinggi; i++) {      let baris = '';      if (i === 1) {        baris = '\*';      } else {        baris = '/';        for (let j = 1; j <= i; j++) {          baris += '\*';        }        baris += '\\';      }      console.log(baris);    }  }  tangga(5); |
| **ScreenShot** |  |

|  |  |
| --- | --- |
| **Soal Nomor** | 3 |
| **Jawaban Soal** | function hapusDuplikat(word) {    let charLastIndex = new Map();    // Store the last index of each character    for (let i = 0; i < word.length; i++) {      charLastIndex.set(word[i], i);    }    // Create an array of characters at their last index, sorted by index    let result = [];    for (let [char, index] of charLastIndex) {      result.push({ char, index });    }    result.sort((a, b) => a.index - b.index);    // Extract only the characters    return result.map((item) => item.char).join('');  }  console.log(hapusDuplikat('imajination'));  console.log(hapusDuplikat('associate')); |
| **ScreenShot** |  |

|  |  |
| --- | --- |
| **Soal Nomor** | 4 |
| **Jawaban Soal** | function produk(prices) {    // Menghitung total harga dengan looping    let total = 0;    for (let i = 0; i < prices.length; i++) {      total += prices[i];    }    // Menentukan diskon dan bonus    let discount = 0;    let bonus = '';    if (total > 400000) {      discount = total \* 0.1;      bonus = 'Ransel';    } else if (total > 200000) {      discount = total \* 0.07;      bonus = 'Payung';    } else if (total > 70000) {      discount = total \* 0.05;      bonus = 'Topi';    }    let finalPrice = total - discount;    return `Total Harga = ${total.toFixed(3).replace('.', ',')} Diskon = ${discount.toFixed(3).replace('.', ',')} Bonus = ${bonus} Harga Akhir = ${finalPrice.toFixed(3).replace('.', ',')}`;  }  console.log(produk([2000, 30000, 50000])); |
| **ScreenShot** |  |

|  |  |
| --- | --- |
| **Soal Nomor** | 5 |
| **Jawaban Soal** | function solution(noAkun, nominal) {    let akunUnik = [];    for (let i = 0; i < noAkun.length; i++) {      let sudahAda = false;      for (let j = 0; j < akunUnik.length; j++) {        if (akunUnik[j] === noAkun[i]) {          sudahAda = true;          break;        }      }      if (!sudahAda) {        akunUnik.push(noAkun[i]);      }    }    let hasil = [];    let totalDebit = 0;    let totalKredit = 0;    for (let i = 0; i < akunUnik.length; i++) {      let akun = akunUnik[i];      let debit = 0;      let kredit = 0;      for (let j = 0; j < noAkun.length; j++) {        if (noAkun[j] === akun) {          if (j === 0) {            debit += nominal[j];          } else {            kredit += nominal[j];          }        }      }      totalDebit += debit;      totalKredit += kredit;      hasil.push([akun, debit, kredit]);    }    console.log('#Akun # Debit # Kredit');    for (let i = 0; i < hasil.length; i++) {      console.log(`${hasil[i][0]}-${hasil[i][1]}-${hasil[i][2]}`);    }    let status = totalDebit === totalKredit ? 'Balance' : 'Not Balance';    console.log(`#Total #${totalDebit} #${totalKredit} (${status})`);  }  // Tes kasus 1  let noAkun1 = [111, 211, 201];  let nominal1 = [200000, 50000, 150000];  solution(noAkun1, nominal1);  // Tes kasus 2  let noAkun2 = [111, 201];  let nominal2 = [100000, 120000];  solution(noAkun2, nominal2); |
| **ScreenShot** |  |

# Soal Database

|  |  |
| --- | --- |
| **Soal Nomor** | 1 |
| **Jawaban Soal** | INSERT INTO Header\_Transaksi (transaction\_id, customer\_name, transaction\_date)  VALUES  ('Trx001', 'Rivaldo', '2022-10-22'),  ('Trx002', 'Melina', '2023-04-07');  INSERT INTO Detail\_Transaksi (transaction\_id, product\_name, price, qty, discount)  VALUES  ('Trx001', 'Pan Pizza', 30000.00, 1, 5.0),  ('Trx001', 'Crown Crust', 45000.00, 1, 5.0),  ('Trx002', 'Cheesy Bites', 47000.00, 2, 10.0); |
| **ScreenShot** |  |

|  |  |
| --- | --- |
| **Soal Nomor** | 2 |
| **Jawaban Soal** | **UPDATE** Detail\_Transaksi  **SET** discount = **CASE**  **WHEN** transaction\_id = 'Trx001' **AND** product\_name = 'Pan Pizza' **THEN** 0.025  **WHEN** transaction\_id = 'Trx001' **AND** product\_name = 'Crown Crust' **THEN** 0.075  **WHEN** transaction\_id = 'Trx002' **AND** product\_name = 'Cheesy Bites' **THEN** 0.15  **ELSE** discount  **END**  **WHERE** transaction\_id **IN** ('Trx001', 'Trx002'); |
| **ScreenShot** |  |

|  |  |
| --- | --- |
| **Soal Nomor** | 3 |
| **Jawaban Soal** | **SELECT**  ht.transaction\_id **AS** 'ID',  ht.customer\_name **AS** 'Customer',  ht.transaction\_date **AS** 'Date',  **SUM**(dt.qty) **AS** 'Qty',  **SUM**(dt.price \* dt.qty) **AS** 'Price',  **ROUND**(**SUM**(dt.price \* dt.qty \* dt.discount), 0) **AS** 'Total Discount',  **ROUND**(**SUM**(dt.price \* dt.qty \* (1 - dt.discount)), 0) **AS** 'Total Price'  **FROM**  Header\_Transaksi ht  **JOIN**  Detail\_Transaksi dt **ON** ht.transaction\_id = dt.transaction\_id  **GROUP** **BY**  ht.transaction\_id, ht.customer\_name, ht.transaction\_date  **ORDER** **BY**  ht.transaction\_id; |
| **ScreenShot** |  |