**PCP(Personal Contract Purchase) Calculator in React JS: Case Study**

**Background:**

John, an individual living in the UK, is considering purchasing a car. To assess his monthly payments and other costs, he turns to a website that provides a PCP Calculator built with React JS.

**Functional Requirements:**

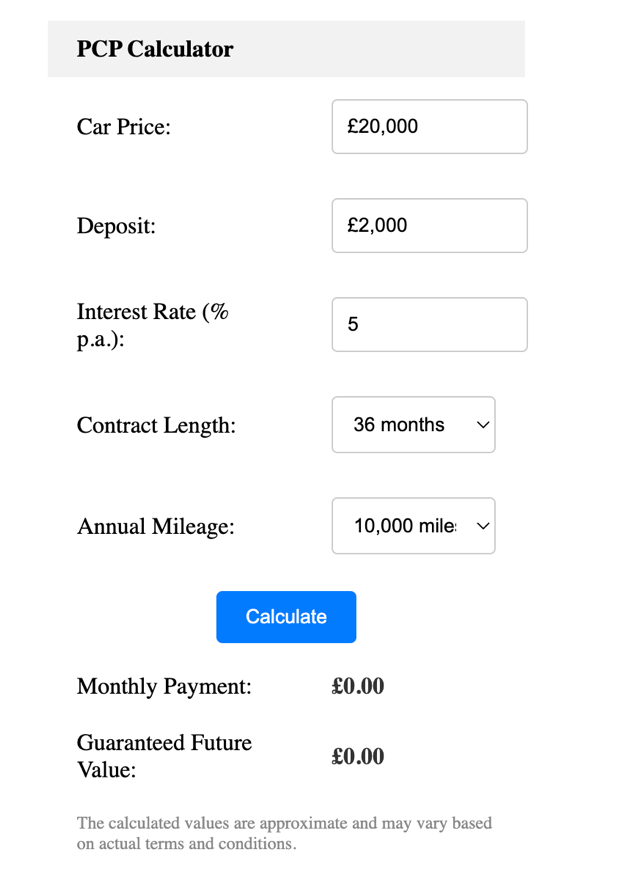
1. **User Interface:**
   * **Car Price Input**: A text input where the user can enter the total price of the car.
   * **Deposit Input**: A text input where the user can enter the deposit amount they intend to pay upfront.
   * **Interest Rate Input**: A text input where the user can enter the interest rate.
   * **Contract Length Input**: A dropdown or slider to select the length of the contract in months.
   * **Estimated Mileage Input**: A dropdown or slider to choose the estimated annual mileage.
   * **Calculate Button**: A button that, when pressed, will calculate the monthly payments and GFV.
   * **Output Area**: An area to display the monthly payment and GFV after the calculation.

**PCP Calculator: Mock UI Description**

1. **Title Area**:
   * **Header**: "PCP Calculator"
2. **Input Area**:
   * **Car Price**: Label "Car Price" with an input box next to it. Pre-filled with "£20,000".
   * **Deposit**: Label "Deposit" with an input box next to it. Pre-filled with "£2,000".
   * **Interest Rate**: Label "Interest Rate (per annum)" with an input box next to it (with a "%" symbol at the end). Pre-filled with "5".
   * **Contract Length**: Label "Contract Length" with a dropdown next to it. Options: "12 months", "24 months", "36 months", etc. Default selected is "36 months".
   * **Estimated Annual Mileage**: Label "Annual Mileage" with a dropdown next to it. Options like "5,000 miles", "10,000 miles", "15,000 miles", etc. Default selected is "10,000 miles".
   * **Calculate Button**: A prominently visible button labeled "Calculate".
3. **Output Area**:
   * **Monthly Payment**: Label "Monthly Payment" with a large, bold number next to it (to be filled after calculation). Placeholder: "£0.00".
   * **Guaranteed Future Value**: Label "Guaranteed Future Value" with a large, bold number next to it (to be filled after calculation). Placeholder: "£0.00".
4. **Footer**:
   * Some disclaimers or notes like "The calculated values are approximate and may vary based on actual terms and conditions."

**Styling and UX Notes**:

* The entire calculator can be encased in a card-like container with rounded corners and subtle shadow for depth.
* Use a clean, legible font like "Roboto" or "Arial".
* Input fields should have a consistent style: slight border-radius, comfortable padding, and a subtle border.
* The "Calculate" button can be in a contrast color to make it stand out. For example, if the overall theme is light, the button could be a moderate blue with white text.
* The output values (Monthly Payment and GFV) should be in larger font size and perhaps a different color to make them stand out from the rest of the content.
* Ensure the UI is responsive: On smaller screens, elements might need to stack vertically, while on larger screens, they can be in a more spread-out, grid-like arrangement.



1. **Functionality**:
   * **GFV Calculation**: The application should use the car price, estimated mileage, and contract length to estimate the GFV. (For simplicity, you can use a basic formula, though in a real-world scenario, this would be more complex and based on various depreciation rates and factors.)
   * **Interest Calculation**: Calculate the total interest over the term of the contract based on the interest rate.
   * **Monthly Payment Calculation**: As per the formula mentioned earlier.
   * **Validation**: Ensure that all inputs are valid numbers and that essential inputs are not left empty.
2. **User Experience**:
   * **Error Handling**: If the user enters an invalid input (e.g., a negative number or text in a number field), the application should show an error message.
   * **Responsive Design**: The calculator should be usable on both desktop and mobile devices.
   * **Result Highlight**: When the calculation is completed, the results should be highlighted or emphasized in some way, so they stand out to the user.

**Technical Requirements:**

1. **State Management**: Use React's useState and useContext (if needed) for managing the state of the inputs and results.
2. **Components**: Break down the application into reusable components. For example:
   * InputComponent: To be used for various inputs.
   * ResultComponent: To display the calculated results.
   * CalculatorComponent: To encapsulate the entire calculator functionality and layout.
3. **Styling**: Utilize CSS modules or styled-components for styling. Ensure the design is clean and intuitive.
4. **Testing**: Implement unit tests using libraries like Jest and React Testing Library to test the components and calculation logic.

**Sample User Journey:**

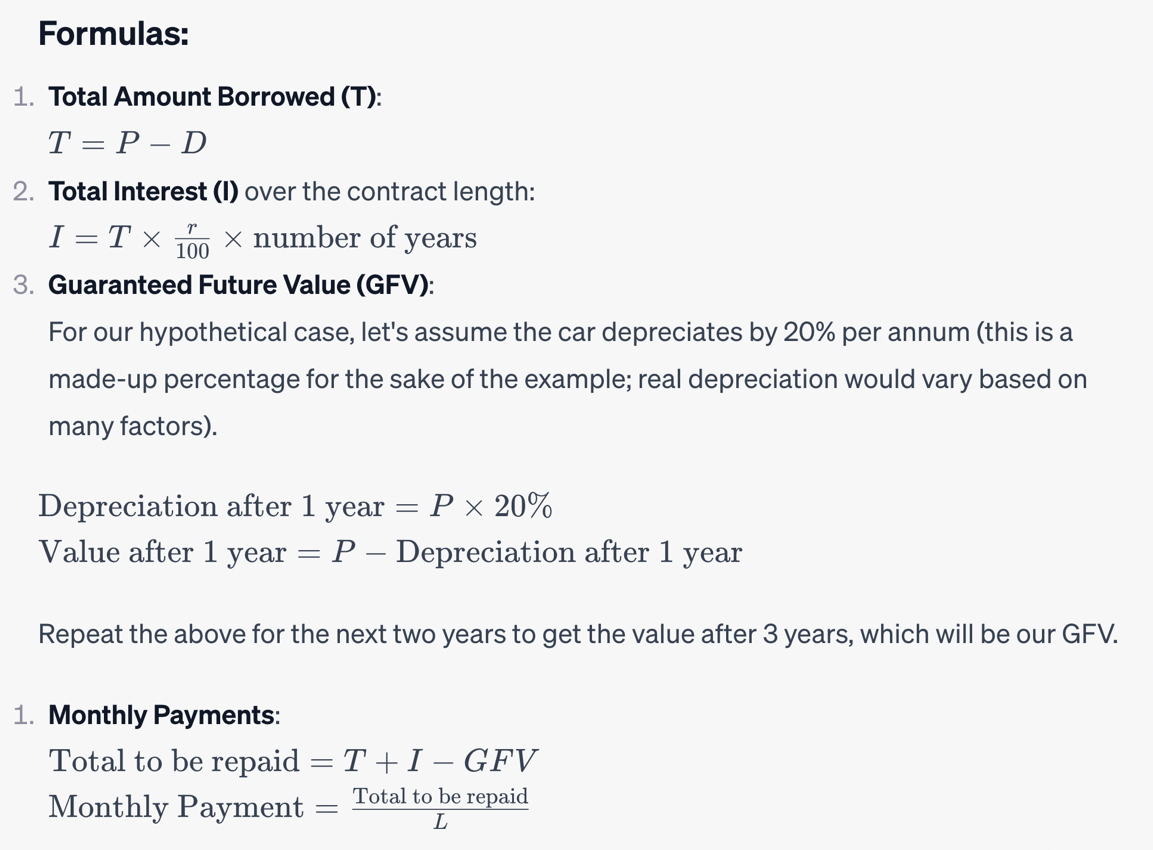
1. John visits the PCP Calculator website.
2. He enters the car price of £20,000 and an upfront deposit of £2,000.
3. He then sets the interest rate to 5% and selects a contract length of 36 months with an estimated annual mileage of 10,000 miles.
4. Upon pressing the "Calculate" button, the calculator displays a monthly payment of (hypothetically) £350 and a GFV of £9,000.
5. John now has a clearer understanding of his potential monthly costs and the balloon payment he might need to make if he decides to keep the car.

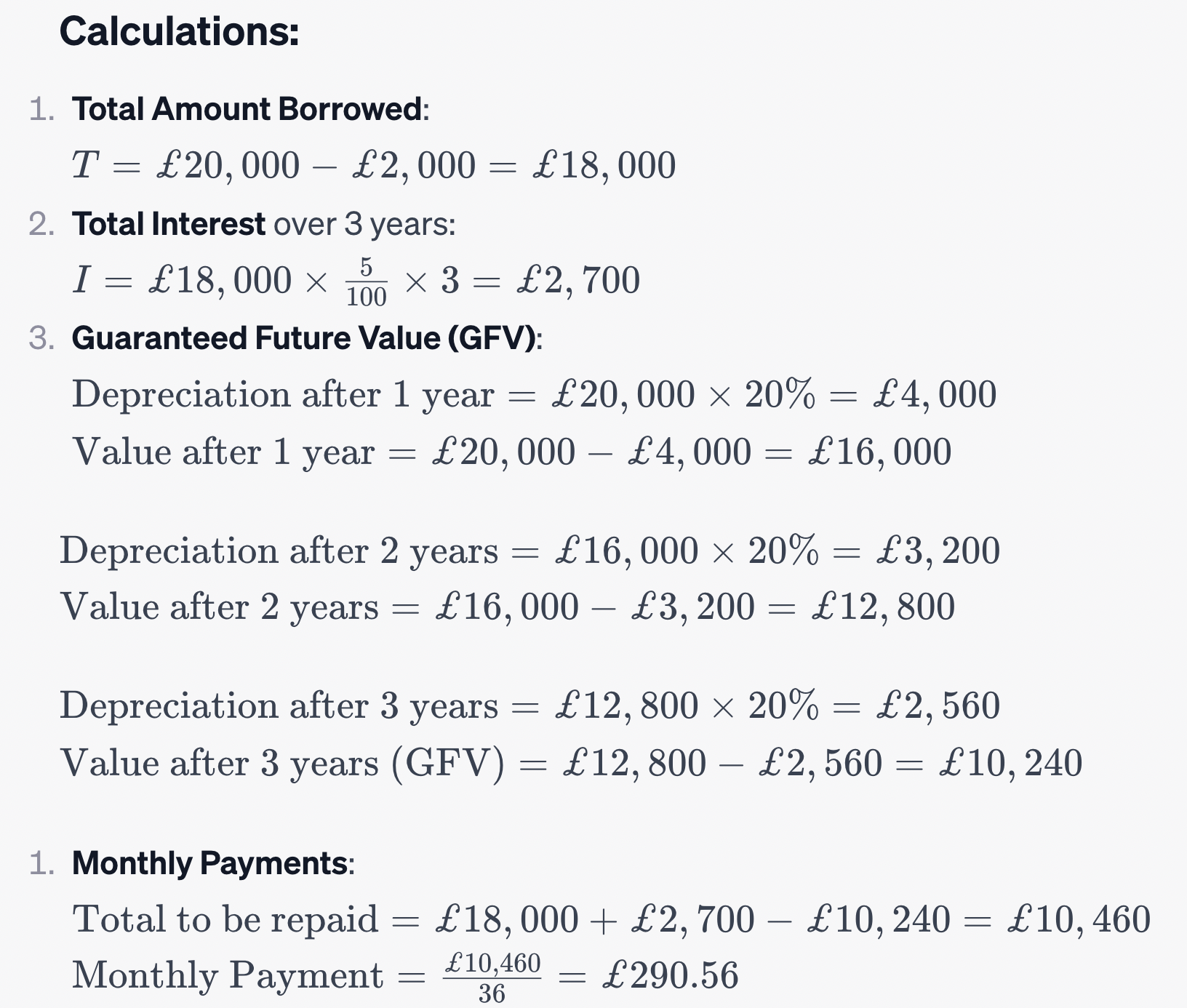
Break down of formulas and calculations based on the scenario provided.

Refer <https://money.stackexchange.com/questions/131849/formula-for-calculating-the-monthly-payment-for-a-personal-contract-purchase>

**Variables from the Scenario:**

* **Car Price (P)**: £20,000
* **Deposit (D)**: £2,000
* **Interest Rate (r)**: 5% per annum
* **Contract Length (L)**: 36 months or 3 years
* **Estimated Annual Mileage**: 10,000 miles (For simplicity, we'll not use mileage in our basic calculations here, but in reality, mileage affects the depreciation and therefore the GFV.)





So, John would pay approximately £290.56 per month, and if he wants to keep the car at the end of the 3-year period, he would pay the GFV of £10,240.