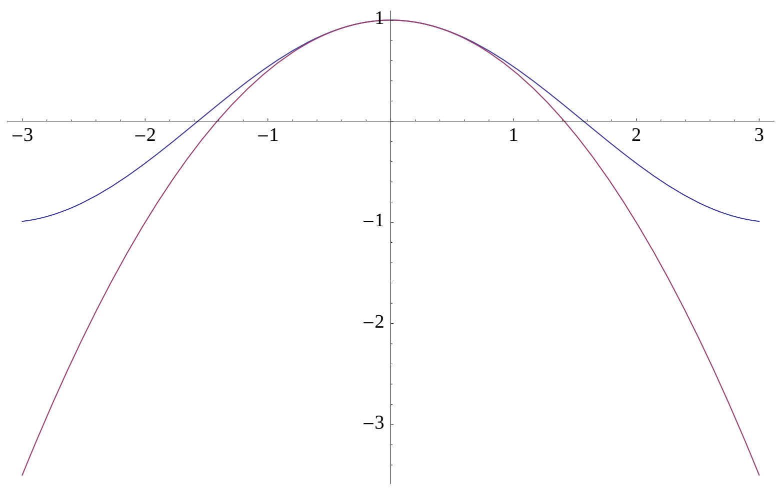
```
(* Taylor Polynomials | Laboratory Project | Mr. Shubleka *)

(* 1 *)

f[x_] := Cos[x];

P[x_] := A + B * x + C * x^2;

Solve[f[0] == P[0], A]

{{A → 1}}

Solve[P'[0] == f'[0], B]

{{B → 0}}

Solve[P''[0] == f''[0], C]
```

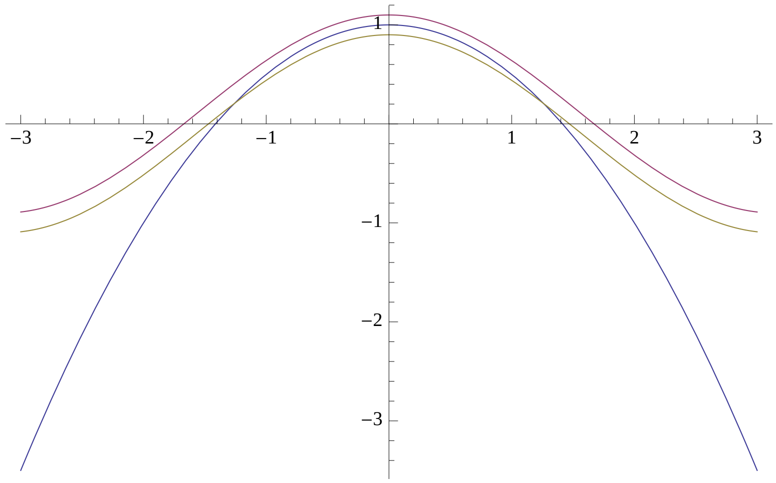
$\left\{\left\{C \to -\frac{1}{2}\right\}\right\}$

```
P[x_] := 1 - 0.5 x^2;

Plot[{f[x], P[x]}, {x, -3, 3}]
```




```
(* 2 *)
```

```
Plot[{P[x], f[x] + 0.1, f[x] - 0.1}, {x, -3, 3}]
```



x ~~ ±1.26124

```
(* 3 *)
```

P (a) = f (a)
f '(x) = B + 2 C (x - a), so then f '(a) = B
f ''(x) = 2 C, so then f ''(a) = 2 C, which makes C = f ''(a)/2

```
(* 4 *)

f[x_] := Sqrt[x + 3];

A = f[1]
```

2

```
B = f'[1]
```

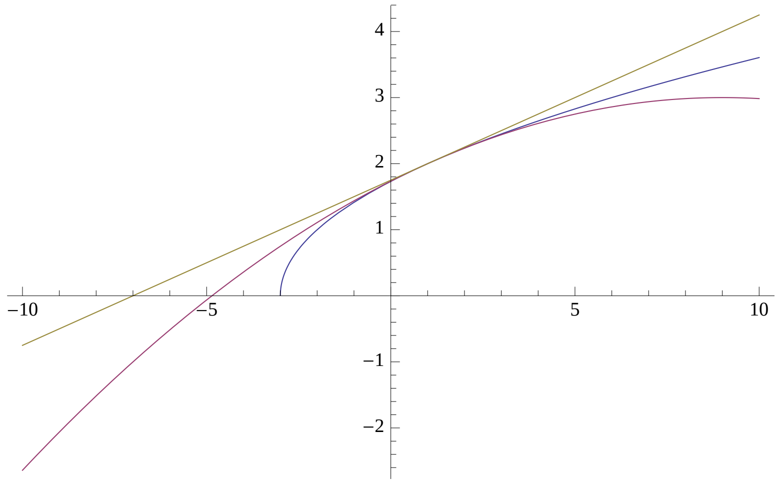
$\frac{1}{4}$

```
CD = f''[1] / 2
```

$-\frac{1}{64}$

```
L[x_] := A + B (x - 1);

P[x_] := A + B (x - 1) + CD (x - 1)^2;
```

```
Plot[{f[x], P[x], L[x]}, {x, -10, 10}]
```

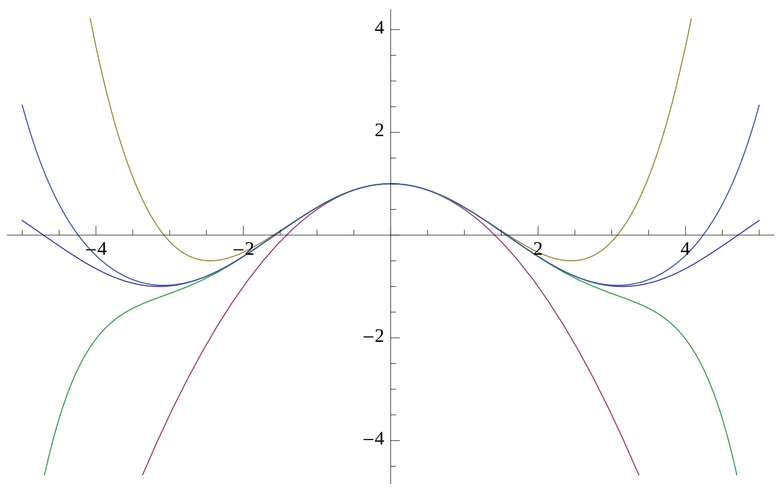


Conclusion : The quadratic function is a better approximation near x = 1.

```
(* 5 *)
```

Plug in x = a.
Differentiate, plug in x = a.
Differentiate again, plug in x = a... repeat to find a pattern for the n-th coefficient.

```
(* 6 *)

f[x_] := Cos[x];

T2[x_] := f[0] + f'[0] (x - 0) + f''[0] / 2! (x - 0)^2;
T4[x_] := T2[x] + f'''[0] / 3! (x - 0)^3 + f''''[0] / 4! (x - 0)^4;
T6[x_] := T4[x] + f''''''[0] / 6! (x - 0)^6;
T8[x_] := T6[x] + f''''''''[0] / 8! (x - 0)^8;

Plot[{f[x], T2[x], T4[x], T6[x], T8[x]}, {x, -5, 5}]
```



The higher the degree, the better the approximation. The polynomial of degree 8 "hugs" the graph of cos (x) for a wider range of x - values centered at the origin.