*package main*

*import (*

*"fmt"*

*"net/http"*

*"strconv"*

*)*

*func offsetPagination(w http.ResponseWriter, r \*http.Request) {*

*page, \_ := strconv.Atoi(r.URL.Query().Get("page"))*

*limit, \_ := strconv.Atoi(r.URL.Query().Get("limit"))*

*if page <= 0 {*

*page = 1*

*}*

*if limit <= 0 {*

*limit = 10*

*}*

*offset := (page - 1) \* limit*

*query := fmt.Sprintf("SELECT \* FROM items LIMIT %d OFFSET %d", limit, offset)*

*fmt.Println("Executing query:", query)*

*w.Write([]byte("Offset-based pagination result"))*

*}*

*func cursorPagination(w http.ResponseWriter, r \*http.Request) {*

*cursor := r.URL.Query().Get("cursor")*

*limit, \_ := strconv.Atoi(r.URL.Query().Get("limit"))*

*if limit <= 0 {*

*limit = 10*

*}*

*query := fmt.Sprintf("SELECT \* FROM items WHERE id > '%s' ORDER BY id LIMIT %d", cursor, limit)*

*fmt.Println("Executing query:", query)*

*w.Write([]byte("Cursor-based pagination result"))*

*}*

*func main() {*

*http.HandleFunc("/offset-pagination", offsetPagination)*

*http.HandleFunc("/cursor-pagination", cursorPagination)*

*fmt.Println("Server is running on port 8080")*

*http.ListenAndServe(":8080", nil)*

*}*