#' Get Distance and Travel Time Between Two Locations  
#'  
#' Queries the Google Distance Matrix API to retrieve the distance and estimated travel time  
#' between an origin and a destination. Also retrieves the polyline for the route.  
#'  
#' @param origen A character string specifying the starting location (address or coordinates).  
#' @param destino A character string specifying the destination location (address or coordinates).  
#' @param key A character string with your Google Maps API key. Defaults to `Sys.getenv("API\_KEY")`.  
#' @param time Optional. The departure time to use for the request (either as a POSIXct object or a timestamp).  
#' @param sleep Numeric. Number of seconds to wait after making the API call. Default is `2`.  
#' @param mode Character. Mode of transportation to use. Options include `"driving"`, `"walking"`, `"bicycling"`, and `"transit"`. Default is `"driving"`.  
#' @param ... Additional arguments passed to `googleway::google\_distance()`.  
#'  
#' @return A `tibble` containing:  
#' - `origin\_address`: Resolved origin address.  
#' - `destination\_address`: Resolved destination address.  
#' - `polyline`: Encoded polyline representing the route.  
#' - `distance\_value`: Distance in meters.  
#' - `distance\_text`: Human-readable distance.  
#' - `duration\_value`: Duration in seconds.  
#' - `duration\_text`: Human-readable duration.  
#'  
#' @details  
#' - The function uses `googleway::google\_distance()` to obtain distance and duration data,   
#' and `googleway::google\_directions()` to retrieve the polyline.  
#' - The result is printed as a JSON string and returned as a tibble.  
#' - A sleep is enforced after the API call to respect API usage limits.  
#'  
get\_distance\_time <- function(  
 origen,  
 destino,  
 key = Sys.getenv("API\_KEY"),  
 time = NULL,  
 sleep = 2,  
 mode = "driving",  
 ...  
) {  
 query\_result <- googleway::google\_distance(  
 origins = origen,  
 destinations = destino,  
 key = key,  
 mode = mode,  
 departure\_time = time,  
 ...  
 )  
   
 result <- query\_result$rows$elements |> as.data.frame()  
   
 rsult\_data <- tibble::tibble(  
 distance\_value = result$distance$value,  
 distance\_text = result$distance$text,  
 duration\_value = result$duration$value,  
 duration\_text = result$duration$text  
 )  
   
 polyline <- googleway::google\_directions(origen, destino, key = key) |>   
 googleway::direction\_polyline()  
   
 output <- tibble::tibble(  
 origin\_address = query\_result$origin\_addresses,  
 destination\_address = query\_result$destination\_addresses,  
 polyline = polyline  
 ) |>   
 dplyr::bind\_cols(rsult\_data)  
   
 Sys.sleep(sleep)  
 print(jsonlite::toJSON(output))  
 output  
}