

Programming Constructs -Dictionaries

#### 6. Dictionaries



Dictionary / associative arrays / hash map are very useful data structures

Dictionary are key value map similar to HaspMap in Java or the one in python

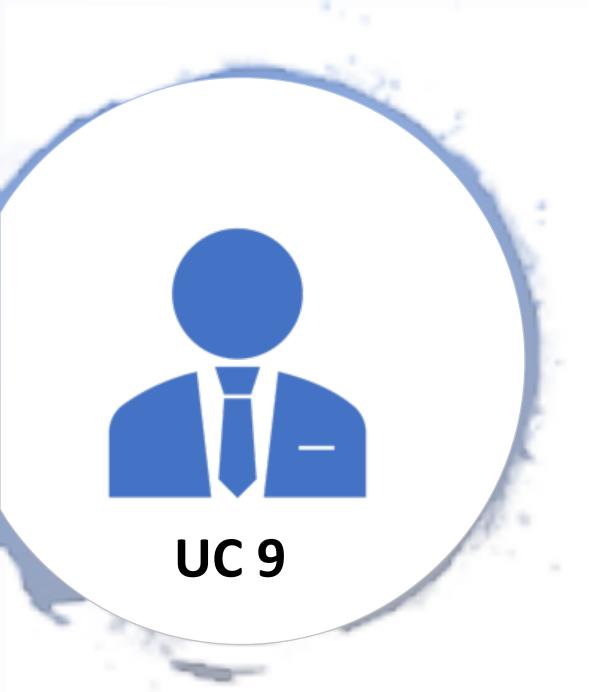
### Dictionary Example

```
#!/usr/local/bin/bash -x
#Note using Latest Bash Version 5.0

declare -A sounds
sounds[dog]="bark"
sounds[cow]="moo"
sounds[bird]="tweet"
sounds[wolf]="howl"

echo "Dog Sound " ${sounds[dog]} # Dog's sound
echo "All Animal Sound " ${sounds[@]} # All values
echo "Animal " ${!sounds[@]} # All keys
echo "Number of Animals " ${#sounds[@]} # Number of elements
unset sounds[dog] # Delete dog
```

```
+ declare -A sounds
+ sounds[dog]=bark
+ sounds[cow]=moo
+ sounds[bird]=tweet
+ sounds[wolf]=howl
+ echo 'Dog Sound ' bark
Dog Sound bark
+ echo 'All Animal Sound ' tweet bark moo howl
All Animal Sound tweet bark moo howl
+ echo 'Animal ' bird dog cow wolf
Animal bird dog cow wolf
+ echo 'Number of Animals ' 4
Number of Animals 4
+ unset 'sounds[dog]'
```



Store the Day and the Daily Wage along with the Total Wage

# Storing Day & Daily Wage

#1/bin/bash -x

```
# CONSTANTS FOR THE PROGRAM
IS_PART_TIME=S;
IS_FULL_TIME=2;
MAX_MRS_IN_MONTHU-4;
EMP_BATE_PER_HR+20;
NUM_WORKING_DAYS=20:
# VARCABLES
tota/worknowrs=0;
totalWorkingDwys=#;
function getWorkingHours() (
   case $5 in
      $15_FULL_TIME)
         work/Noursell
      BIS_PART_TIME:
         work/kours+4
      -
         work/Hours+®
         ==
   echo SworkHours
Function calchailyMage() (
   local workwra-51
   wage=$(($workHrs+$EMP_RATE_PER_HR))
   echo Swage
while [[ StotalMorkHours -10 SMAX HDS IN MONTH &&
         StotalMorkingDays -lt SNUM_MDRKING_DKFS ]]
   ((totalWorkingBays++))
   workHours="$( getWorkingHours $((BANDGHG)) )"
   totalWorkHours+S((StotalWorkHours+SworkHours))
   empOailyMage["StotalMorKingOays"]="$( calcOailyMage SworkMours )"
GIOTAL
totalSalary+"5( calcDuilyWage StotalWorkWours )"
echo "Gaily Wage " $(empGailyWage[0])
echo "All Keys " ${!empDuilyWage[8]3
```

```
+ IS_PART_TIME=1
* IS_FULL_TIME=2
+ MAX_HRS_IN_MONTH+4
* EMP_RATE_PER_HR=28

    NUM_WORKING_DAYS=28

* totalWorkHours=8
+ totalWorkingDays=8
* [[ 0 -lt 4 ]]
* [[ 0 -lt 20 ]]
+ (( totalWorkingDays++ ))
** getWorkingHours 2
++ case $1 in
++ workHours=8
++ echo 8
+ workHours=8
+ totalWorkHours=8
++ calcDailyWage 8
++ local workHrs=8
++ wage=160
++ echo 160
+ empDailyWage["$totalWorkingDays"]=160
+ [[ 8 -lt 4 ]]
** calcDailyWage 8
** local workMrs=8
++ wage=160
** echo 168

    totalSalary=160

* echo 'Daily Wage ' 168
Daily Wage 168
* echo 'All Keys ' 1
All Keys 1
```

### Dictionary Practice Problems



- 1. Write a program in the following steps
  - a. Roll a die and find the number between 1 to 6
  - b. Repeat the Die roll and find the result each time
  - c. Store the result in a dictionary
  - d. Repeat till any one of the number has reached 10 times
  - e. Find the number that reached maximum times and the one that was for minimum times
- Write a Program to generate a birth month of 50 individuals between the year 92 & 93. Find all the individuals having birthdays in the same month. Store it to finally print.



## Thank You