

# নিজে নিজে বেসিক শিখুন (২য় পর্ব)

আসাদুর রহমান

ঢোকানো যাক, যেমন,  
15 XS = "PROCESS"

টাইপ করে <Enter> কী চাপুন। এখন যদি LIST কমান্ডটি Execute করেন তাহলে নতুন লাইনটি ঘোষিত হলে অর্থাৎ 10 ও 20 নম্বর লাইনের মাঝেই দেখা যাবে।

যদি আপনি বর্তমান প্রোগ্রামের শেষে আরেকটি লাইন যোগ করতে চান তাহলে প্রোগ্রামের শেষ লাইন নম্বরের চেয়ে বাও লাইন নম্বর দিয়ে কোমেন্টে শুরু করুন। 40 PRINT X S টাইপ করে <Enter> কী চাপুন। এখন 20 নম্বর লাইন পরিবর্তিত হয়েছে এবং 15 এর 40 নম্বর লাইন আগের প্রোগ্রামে যোগ হয়েছে। এবার Run কমান্ড ব্যবহার করে প্রোগ্রামটি Execute করুন।

আইডেন্টিটি হবে :  
Very Good  
PROCESS  
Ok

আপনি যদি প্রোগ্রামের কোন লাইন মুছে ফেলতে চান তাহলে শুধু ঐ লাইন নম্বর টাইপ করে <Enter> কী চাপুন; লাইনটি মুছে যাবে। 140 দিয়ে এটার চাপলে বর্তমান প্রোগ্রাম থেকে 40 নম্বর লাইনটি মুছে যাবে। আপনি নিশ্চিত হবার জন্য প্রোগ্রামকে LIST বা Execute করে দেখুন।

এ পর্যন্ত আমরা LET স্টেটমেন্টে শুধু কনস্ট্যান্ট (ডাটা) ব্যবহার করে এসেছি। প্রোগ্রামে ডাটাতুলো অপরিচিত ছিল। ফলে প্রোগ্রাম ঐ নির্দিষ্ট ডাটা নিয়েই কাজ করছে এবং নতুন ডাটা নিয়ে কাজ করতে হলে প্রোগ্রাম স্টেটমেন্টকে পরিবর্তন করতে হতো। উদাহরণ হলো :

PRINT 1024\*5/100

স্টেটমেন্টটি সর্বসময় 1024 এর শতকরা 5 কত তাই দেখাবে, কারণ এখানে 1024 ও 5 দুটোই কনস্ট্যান্ট। শুধুমাত্র কনস্ট্যান্ট ডাটা ছাড়া ভিন্ন ভিন্ন ধরনের ডাটা দেয়া যায়। অন্যদু উপায়ে মধ্যে INPUT স্টেটমেন্ট অন্যতম। এর Syntax নিচ দেয়া হলো— INPUT ["Prompt"], Variable [, Variable, Variable, ...] এটি একটি স্টেটমেন্ট

INPUT স্টেটমেন্টে কমান্ডের একটি ভেরিয়েবল ACSJ ই থাকতে হবে। যেমন, INPUT A, INPUT C, ইত্যাদি। একজন অপারেটরের কাছ থেকে ডাটা গ্রহণ করার জন্য প্রোগ্রামে INPUT স্টেটমেন্টটি ব্যবহার হয়। আগে থেকেই কনস্ট্যান্ট ডাটা ব্যবহার না করে, প্রতিবারে Execute করার সময় অপারেটরের ইচ্ছানুযায়ী ডাটা ব্যবহার করা যায়। ফলে ভিন্ন ভিন্ন ডাটা ব্যবহার করে প্রোগ্রামটি কাজ লাগানো হবে।

যখনই বেসিক প্রোগ্রামে INPUT স্টেটমেন্টটি পায় তখন এটা প্রোগ্রাম Execution বন্ধ করে একটি প্রশ্নাবলী চাই (?) দেয়ায়। এর দ্বারা আমরা বুঝতে পারি বেসিক ডাটা এবং তারপর একটি <Enter> কী চাপার অপেক্ষা আছে। যদি আমরা ইনপুটের সাথে Prompt (শিউ) ব্যবহার করি তাহলে বেসিক ঐ Prompt প্রশ্নাবলীকে চিহ্নিত করে দেখাবে। আপনি একটি INPUT স্টেটমেন্টে একাধিক ভেরিয়েবল এবং একই প্রোগ্রামে একাধিক INPUT স্টেটমেন্ট ব্যবহার করতে পারেন। নিউমেরিক ডাটার মান নিউমেরিক ভেরিয়েবল এবং স্ট্রিং ডাটার মান স্ট্রিং ভেরিয়েবল INPUT এর পরে ব্যবহার করতে হবে। আপনি ইনপুট ডাটা হিসাবে যে সকল কাঠেরের টাইপ করবেন সেগুলো স্ট্রিং হিসাবে দেখাবে এবং <Enter> কী চাপলে সেগুলো প্রোগ্রামের ভেতর ঢুকবে। INPUT স্টেটমেন্ট যে ভেরিয়েবল নির্দিষ্ট করা আছে সেটিতে আপনার টাইপ করা ডাটা Store হবে। এর পর বেসিক উভয় নম্বরের লাইনগুলো Execute করে যাবে।

INPUT স্টেটমেন্টে ডাটাবাহার বোঝার জন্য নিচের প্রোগ্রামটি লিখে Run করুন :

কিন্তু বেসিক সর্বসময়ই LIST কমান্ড Execute করার পর Direct mode -এ ফিরে যায় এবং Ok Message দেখে। LIST কে বন্ধ করার জন্য Ctrl এবং Break কী (Run/Stop কী) চাপা যোগে পারে। List কমান্ডের পর লাইন নম্বর এবং যাইহোক (-) দেয়া হলে সেই নম্বরের লাইন এবং তার পরেরটি নম্বরযুক্ত সব লাইনগুলো স্ট্রিং হিসাবে আসবে। LIST কমান্ডের উদাহরণঃ

কীওয়ার্ড	আওতমেন্ট	কাজ
LIST		পুরো প্রোগ্রাম List হবে।
LIST 10		শুধুমাত্র লাইন 10 List হবে।
LIST 230-		লাইন 230 সহ পরের সব লাইন List হবে।
LIST -230		তত থেকে লাইন 230 পর্যন্ত List হবে।
LIST 55-155		লাইন 55 থেকে লাইন 155 পর্যন্ত List হবে।

পুরো প্রোগ্রামটি স্ট্রিং হিসাবে List টাইপ করে <Enter> কী চাপুন। লাইন 20 কে এন্ট্রি করার চেষ্টা করুন। Constant 5 কে 3.4 এ পরিবর্তন করতে হবে। কার্সরকে 5 এর নিচে আনুন, তারপর 3.4 টাইপ করে <Enter> কী চাপুন। এখন কার্সরকে নিচে একটা ফাঁকা লাইনে নিয়ে এসে RUN টাইপ করে <Enter> কী চাপুন অথবা F2 ফাংশন কী চাপুন। আপনার নতুন পরিবর্তিত—এর প্রোগ্রামটি Execute করবে। আউটপুট হবে 3.4।

নিচের প্রোগ্রামটি লিখুন। এ প্রোগ্রামটি দুটি শিউ ভেরিয়েবলে দুটি শিউ কনস্ট্যান্ট রাখে এবং আউটপুট ভেরিয়েবলগুলো মান দেখায়।

নতুন প্রোগ্রাম লিখার আগে NEW কমান্ডটি ব্যবহার করার কথা আবারো মনে করিয়ে দিচ্ছি।

```

প্রোগ্রাম 1খ
NEW
10 LET AS = " Very"
20 LET BS = " Nice Day"
30 ? AS + BS
RUN
Very Nice Day
Ok
    
```

এ উদাহরণে 10 নম্বর লাইন ব্যবহার হয়েছে শিউ ভেরিয়েবল AS এ Very শিউটি রাখার জন্য এবং 20 নম্বর লাইন ব্যবহার হয়েছে শিউ ভেরিয়েবল BS এ Nice Day শিউটি রাখার জন্য। 30 নম্বর লাইনটি ব্যবহার হয়েছে AS ও BS এর বর্তমান মানকে পাশাপাশি দেখে দেখার জন্য।

আপনি যদি প্রোগ্রামের কোন একটি লাইন বদলাতে চান তাহলে সেই লাইন নম্বরটি লিখুন তারপর নতুন স্টেটমেন্ট টাইপ করে <Enter> কী চাপুন। বেসিক হুবহু ঐভাবে লাইনটিকে লাইন নম্বরের উল্লিখিতমানসূত্রে সক্রিয় নেবে এবং নতুন লাইনটি প্রোগ্রামে প্রতিস্থাপিত করবে। 20 নম্বর লাইনটি বদলান করা যাক :

```

20 LET BS = " Good"
তারপর প্রোগ্রামটি Execute করুন।
    
```

আপনি যদি আগের প্রোগ্রামের দুটি লাইনের মাঝে আরেকটি লাইন ঢোকাতে চান তাহলে এখন একটি লাইন নম্বর দিয়ে শুরু করতে হবে যাতে নম্বরটি আগের দুটি লাইন নম্বরের মাঝের কোন একটি নম্বর হয়। 10 ও 20 নম্বর লাইনের মাঝে একটি লাইন

গত সংখ্যায় আমরা বেসিক ডাটা সম্পর্কে সর্বাঙ্গীর্ণ আলোচনা করেছি। এতে বেসিকের কীওয়ার্ডগুলোর মধ্যে PRINT, LET এবং RUN কীওয়ার্ডগুলোর ব্যবহার করার নিয়ম আলোচনা করা হয়েছিল। এ সংখ্যায় আমরা কীওয়ার্ড ও নতুন কিছু কীওয়ার্ড ব্যবহার করে কয়েকটা দুর্ভাবজনক প্রোগ্রাম লিখবো। নতুন প্রোগ্রাম লিখার আগে NEW কমান্ডটি ব্যবহার করুন। এতে যদি আগে কোন প্রোগ্রাম সৃষ্টিতে ত্রুটি থাকে, তবে লাইন নম্বরের সাথে নতুন প্রোগ্রামের লাইন নম্বর মিলে যাবে না। প্রতি স্টেটমেন্টের শেষে <Enter> বা <Return> কী চাপুন।

নিচের প্রোগ্রামগুলো লিখে Run (Execute) করে দেখুন।

```

প্রোগ্রাম 1ক
NEW
10 LET A = 1000
20 LET RATE = 5
30 PRINT A * RATE / 100
RUN
50
Ok
    
```

প্রোগ্রাম 1ক—এ 10 নম্বর লাইনে সাংখ্যিক (Numeric) ভেরিয়েবল A কে 1000 মান দেয়া হয়েছে। অনুপাতভারে 20 নম্বর লাইনে সাংখ্যিক ভেরিয়েবল RATE কে 5 মান দেয়া হয়েছে। 30 নম্বর লাইনটি 1000 এর 5% কত হবে তা বের করার জন্য ব্যবহার করা হয়েছে। একই কাজের জন্য আমরা সর্বাঙ্গীর্ণকার DIRECT MODE এ লিখতে পারি। যথা

```

PRINT 1000*5/100
50
Ok
CLS
    
```

এটা একটি স্টেটমেন্ট DOS এর CLS কমান্ডটি যে কাজ করে বেসিকের CLS স্টেটমেন্টটি ঠিক একই কাজ করে। এ স্টেটমেন্ট স্ট্রিংকো পরিষ্কার করে কার্সরকে স্ট্রিংয়ের ঐদিক এগিয়ে পেরে (HomePosition) নিয়ে যায়। স্ট্রিংকে পরিষ্কার করার জন্য CLS টাইপ করে <Enter> কী চাপুন।

```

CLS
Ok
    
```

বেসিকের LIST কমান্ডটি প্রোগ্রামের লাইনগুলো দেখার কাজে ব্যবহার হয়। LIST টাইপ করে বা F1 ফাংশন কী চাপার পর <Enter> চাপলে আগের লিখা প্রোগ্রামটি স্ট্রিং হিসাবে পাঠবে।

```

LIST
10 LET A = 1000
20 LET RATE = 5
30 PRINT A * RATE / 1000
Ok
    
```

LIST কমান্ডটি সম্পর্কে আরো তথ্যঃ  
LIST [(Line-number)-(Line-number)]—এটি একটি কমান্ড

LIST আপনাকে স্ট্রিং বেসিক প্রোগ্রাম দেখতে দেয়। একবার প্রোগ্রাম স্ট্রিং হিসাবে আনার পর একে খুঁচু সহজে ও দ্রুত এন্ট্রি করা যায়। কোন লাইনে এন্ট্রি করার পর ঐ লাইনে <Enter> কী চাপতে হবে। বর্তমান দেখাচ্ছে আছে এখন প্রোগ্রামের পুরোটা বা অংশ বিশেষ দেখার জন্য LIST কমান্ডটি ব্যবহার করা যায়। প্রোগ্রামের ভেতরেও LIST ব্যবহার করা যাবে,

### প্রোগ্রাম ২৯

NEW

1) CLS

2) INPUT "Enter Amount : ", AMT

3) INPUT "Enter Rate: ", RATE

4) ? AMT ; "Percent of 'A'; is " ;

A\*RATE/100

5) END

আউটপুট

RUN

Enter Amount : 1234

Enter Rate : 5

5 Percent of 1234 is 61.7

Ok

10 নম্বর লাইনটি CLS স্টেটমেন্ট স্ক্রীন পরিষ্কার করতে ব্যবহার হয়েছে। 20 ও 30 নম্বর লাইন ব্যবহারকারী যাকে Amount ও Rate এর মান প্রোগ্রামে দিতে পারে এজন্য ব্যবহার করা হয়েছে। 4র্থ মান INPUT যে পরিমাণের শতকরা অংশ বের করা হবে তার মান নিয়ে AMT ভেরিবেলে রাখা দ্বিতীয়টি শতকরা হারটি নিয়ে RATE ভেরিবেলে রাখা। 40 নম্বর লাইনটি সর্বসংগত পরিমাণের শতকরা অংশ বের করে সুন্দরভাবে দেখায়। 50 নম্বর লাইনটি প্রোগ্রামের সমাপ্তি বোঝায়। ফলাফল বেসিক END স্টেটমেন্ট পায় তখনই এটি প্রোগ্রাম Execution বন্ধ করে Ok দেখে।

GOTO স্টেটমেন্ট প্রোগ্রামকে তারপরের স্টেটমেন্টে না নিয়ে যে লাইন নম্বর নির্দিষ্ট করা হবে সে লাইনে যেতে দেখা করে, GOTO স্টেটমেন্টের Syntax নিম্নরূপ:

GOTO Line-number-এটি একটি স্টেটমেন্ট যদি আপনি GOTO 160 টাইপ করেন তাহলেই

মোটে তাহলে বেসিক 160 নম্বর লাইন থেকে প্রোগ্রাম (যদি মেমরীতে কোন প্রোগ্রাম থাকে থাকে) Execute শুরু করবে। এ স্টেটমেন্ট কোন প্রোগ্রামকে স্বাভাবিক জরাজেঁদে নির্দিষ্ট করা লাইনের দিকে সরিয়ে দেয়। বসন

15) CLS

16) PRINT "COMPUTER"

17) GOTO 160

এটি Execute করলে বার বার স্ক্রীনে 'COMPUTER' শিরিটি দেখাতেই থাকবে। বেসিক প্রোগ্রামকে তার লাইন নম্বরের উল্লেখসমূহের Execute করতে থাকে, লাইন 150, 160, 170 এবং এরপর Execution বন্ধ হয়ে যাবে যদি এর চেয়ে বড় নম্বরের লাইন না থাকে বা যদি GOTO স্টেটমেন্ট দ্বারা প্রোগ্রামের জরাজেঁদে অন্য কোন লাইনের দিকে নির্দিষ্ট করা না থাকে। কিন্তু এ প্রোগ্রাম 170 নম্বর লাইন Execute করার পর প্রোগ্রাম 160 নম্বর লাইন Execute করবে; যেহেতু এখানে GOTO 160 স্টেটমেন্টটি ব্যবহার হয়েছে। 160 নম্বর লাইনটি Execution শেষ করে 170 নম্বর লাইন Execution হবে এবং তারপরেই প্রোগ্রাম আবার 160 নম্বর লাইনে লাফ দাবে এবং এভাবেই চলতে থাকবে, অর্থাৎ প্রোগ্রাম 160 নম্বর লাইন থেকে 170 নম্বর লাইনে এবং 170 নম্বর লাইন থেকে 160 নম্বর লাইনে যেতে থাকবে বার বার।

অপনার প্রোগ্রামিং-এর জ্ঞানকে সমৃদ্ধ করতে নিজে নিজে আরো প্রোগ্রাম লিখুন। নতুন প্রোগ্রাম লিখার আগে NEW কমান্ডটি ব্যবহার করুন। মনে রাখবেন, প্রোগ্রাম লাইনগুলো লাইন নম্বরের উল্লেখসমূহের Execute করে। অতএব, আপনার প্রোগ্রামের স্টেটমেন্টগুলো সে অনুযায়ী ক্রমশঃ সাজিয়ে লিখবেন। যদি প্রোগ্রাম লিখতে আপনি ভুল করেন, তাহলে Backspace, Del, Insert ইত্যাদি এন্ট্রি

কী ব্যবহার করে প্রোগ্রাম সংশোধন করতে পারেন। ভুল সংশোধন করে <Enter> কী চাপতে হবে।

বেসিক কাজ শেষ হলে SYSTEM কমান্ডটি টাইপ করে <Enter> কী চাপলে লিনা প্রোগ্রাম মেমরী থেকে মুছে যাবে এবং DOS prompt নিজে আসবে। আপানী পূর্বে আরো কীওয়ার্ড এবং বেসিকে এরনে ব্যবহার নিয়ে আলোচনা করবে।

### গত দু'পর্বের সারসর্ম্ম :

দিনের পর দিন ভাল কমপিউটার প্রোগ্রামের ও অপারেশনের গভীরতা বাড়ছে। বেসিক উচ্চ স্তরের জ্ঞানের মধ্যে অন্তর্ভুক্ত গভীরতা ও বহুভাষায় প্রোগ্রামিং ভাষা।

শ্রুতি, ভেরিবেল, কীওয়ার্ড শব্দগুলো কমপিউটার সম্পর্কিত খুব গুরুত্বপূর্ণ শব্দ। সাধারণত কমান্ডের সাথে লাইন নম্বর ব্যবহার হয় না কিন্তু স্টেটমেন্টের আগে লাইন নম্বর ব্যবহার হয়। একটি প্রোগ্রাম হচ্ছে লাইন নম্বরযুক্ত স্টেটমেন্টের একটি সেট। PRINT স্টেটমেন্টটি মনিটর আউটপুট দেখাতে ব্যবহার হয়। কোন ভেরিবেলে মান নির্দিষ্ট (Assign) করতে LET স্টেটমেন্টটি ব্যবহার হয়। RUN কমান্ড নিয়ে বর্তমান প্রোগ্রামটি চলানো যায়। স্ক্রীনে কপি করার জন্য CLS স্টেটমেন্টটি ব্যবহৃত হয়। LIST কমান্ড নিয়ে বর্তমান প্রোগ্রাম লাইনগুলো স্ক্রীনে দেখা যায়। NEW কমান্ডটি মেমরী থেকে বেসিক প্রোগ্রাম মুছে দেয়। কী বোর্ড থেকে পূর্বে রক্ষিত ডাটা গ্রহণ করার জন্য INPUT স্টেটমেন্টটি ব্যবহার হয়। INPUT স্টেটমেন্ট ব্যবহার করলে একটি প্রোগ্রাম ভিন্ন ভিন্ন ভাষা নিয়ে কাজ করতে পারে। END স্টেটমেন্ট প্রোগ্রাম Execution এর সমাপ্তি টাইপ। SYSTEM কমান্ডটি DOS prompt এ আসার জন্য ব্যবহার হয়। আপানী পূর্বে বেসিক সম্পর্কে আরো তথ্য দেখা হবে। (চলবে) অনুশিখন : মোঃ ইয়াসীন ফয়সাল (সুইনি)

## ADMISSION GOING ON COMPUTER COURSES

### PACKAGES

WordStar 4.0, Lotus 1-2-3  
Word Perfect 5.1, dBase III+  
dBase IV, BASIC  
DTP, Mac Draw

### PROGRAMMINGS

dBase, BASIC, Pascal C Programming  
Turbo Pascal  
Assembly Language, PROLOG

### COMPACT COURSES

WordStar 4 & Word Perfect  
WordStar 4 & dBase III+

WS/WP, Lotus 123 & dBase III+

### SPECIAL COURSES FOR

Publishers & Journalists  
Accountants & Bankers  
Administration & Management

BANGLADESH COMPUTER ACADEMY

323/C, Tongi Diversion Road  
Maghbazar Chowrashta,  
DHAKA. Tel: 415648, 415506.



ত্রিশ বছর যাবৎ এদেশে বিজ্ঞানকে  
জনপ্রিয় করতে পথিকৃৎ মাসিক

# বিজ্ঞান সাময়িকী

## পড়ুন

৩৭/সি, আসাদ এভেন্যু

মোহাম্মদপুর, ঢাকা

ফোন : ৮১১৮৯৮